Automatic Identification System data implementation to a Weather Ship Routing

Treball Final de Grau

Facultat de Nàutica de Barcelona
Universitat Politècnica de Catalunya

Treball realitzat per:

Xavier Calvo Surià

Dirigit per:

Marcel·la Castells i Sanabra
Manuel Grifoll Colls

Grau en Nàutica i Transport Marítim

Barcelona, data 10 de Juny de 2020

Departament de Ciència i Enginyeria Nàutiques

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat de Nàutica de Barcelona

Acknowledgements

On the very outset of this dissertation, I would like to extend my sincere and heartfelt obligation towards all the people who have helped me in this endeavour. Without their active guidance, help, cooperation, and encouragement, I would not have made headway in the project.

I am extremely thankful and pay my gratitude to my faculty guides Mrs. Marcel·la Castells I Sanabra and Mr. Manel Grifoll Colls for their valuable guidance and support for completion of my university degree and this project.

I extend my gratitude to *Barcelona School of Nautical Studies*, Mr. Xavier Martínez de Osés, Mrs. Clara Borén, and every other professor for giving me this opportunity.

I also acknowledge with a deep sense of reverence, my gratitude towards my parents and members of my family, who have always supported me morally as well as economically.

A very special feeling of gratitude and love to my girlfriend Andrea who has accompanied me throughout this final step of my education and has supported me to outdo myself in life.

At last but not least, gratitude goes to all of my friends who directly or indirectly helped me to complete this project report.

Thanking you

Xavier Calvo Surià

Resum

Aquesta contribució forma part d'un treball de recerca de la Facultat de Nàutica de Barcelona relacionat amb la optimització de rutes, ampliant i millorant-lo amb la incorporació de tres noves eines. L'algoritme, anomenat SIMROUTE, ha estat desenvolupat per un grup de recerca de la UPC i inicialment comparava la ruta òptima amb la ruta de mínima distància. L'objectiu principal d'aquest Treball Final de Grau és introduir i analitzar les rutes reals dels vaixells, amb les dades obtingudes a partir del Sistema d'Identificació Automàtica (AIS), i comparar-les amb rutes òptimes obtingudes amb l'algoritme. Per altra banda, el programari de *ship routing* ha guanyat importància recentment degut a l'interès en reduir costos i emissions a Europa. Aquesta treball obre el camí a noves millores en el programari SIMROUTE i noves funcions per a una millor implementació de dades AIS.

Abstract

This contribution belongs to a research work by the Barcelona School of Nautical Studies related to the optimization of maritime routes. Three new tools are introduced to improve and extend it. The algorithm is called SIMROUTE and was developed by a *Universitat Politècnica de Catalunya* (UPC) team, it initially compared the optimum route with a minimum distance route. The objective of this End of Degree Project (TFG) is introduce and analyse real ship routes, with data obtained from the Automatic Identification System (AIS), and compare them with optimum routes obtained with the algorithm. On the other hand, ship routing systems have become more important in recent years since there is a higher interest in reducing costs and fuel emissions in Europe. This investigation gives way to new improvements in the SIMROUTE software for new functions in AIS data implementation.

Index

ACKNOWLEDGEMENTS II
RESUM III
ABSTRACT IV
INDEX V

CHAPTER 1. INTRODUCTION 1

1.1. STATE-OF-THE-ART 1
1.2. OBJECTIVES 2

CHAPTER 2. METHODS 3

2.1. CURRENT SIMROUTE 3
2.2. AIS AS A TOOL FOR SHIP WEATHER ROUTING 4
2.3. WEATHER FORECASTS 6
2.4. NEW INTRODUCTIONS TO SIMROUTE 9
2.5. UPDATE OF SIMROUTE'S USER'S MANUAL 15
2.6. CASE STUDY 18

CHAPTER 3. RESULTS 19

3.1. CASE 1 29/01/2020 TO 18/02/2020 19
..... 22
3.2. CASE 2 30/12/2020 TO 25/01/2020 23
..... 26
3.3. CASE 3 24/01/2020 TO 11/02/2020 27
..... 30
3.4. CASE 4 05/01/2020 TO 28/01/2020 31

CHAPTER 4. DISCUSSION 35

4.1. *FIND_PORTS.M* MODULE 35
4.2. *AIS_ROUTE_ANALYSIS.M* MODULE 35
4.3. *COMPARE_ROUTES.M* MODULE 35
4.4. CASE 1 29/01/2020 TO 18/02/2020 36
4.5. CASE 2 30/12/2020 TO 25/01/2020 38

4.6. CASE 3 24/01/2020 TO 11/02/2020... 40
4.7. CASE 4 05/01/2020 TO 28/01/2020... 42

CHAPTER 5. CONCLUSIONS 44

REFERENCES... 46

ANNEX 1. AIS_ROUTE_ANALYSIS.M 48

ANNEX 2. FIND_PORTS.M 50

ANNEX 3. COMPARE_ROUTES.M 52

ANNEX 4. SIMROUTE USER'S AND TECHNICAL MANUALS 55

ANNEX 5. AIS DATA IMPLEMENTATION TO A WEATHER SHIP ROUTING. IAMUS 2018 XAVIER CALVO SURIÀ'S PAPER.. 57

Chapter 1. Introduction

1.1. State-of-the-art

In the recent years, there has been a huge amount of research on ship routing systems and their applications. Together with 5G, unmanned vessels, and new feasible energy sources, it is taking the ship industry to a new era of modern, safer, and greener navigation.

'Ship weather routing' is defined by the International Maritime Organization (IMO) as the type of ship routing that provides navigators with 'optimum routes' to avoid bad weather. [8] That is to propose safe and efficient routes and, at the same time, consider weather thresholds or the nature of the cargo. However, one of the main things that ship routing is trying to improve is fuel efficiency. According to Vessel Performance Optimisation Magazine, (VPO), 'weather, including currents, waves, wind, and swell has been found to affect the performance of a vessel by between 50 and 80 per cent and cost thousands of euros in additional fuel consumption.' [22] The constant necessity to improve energy efficiency, economic feasibility, and safety, while also complying with emission regulations, is a big motivation for on-going developments and improvements. However, there need to be ways of assessing the condition of the final product since the quality of the provided outcome route not only depends on the quality of the data, but also on the modelling of the optimization problem and algorithm chosen to solve it. [12]

Ship routing uses pathfinding to optimize routes. 'PPathfinding is the plotting, by a computer application, of the shortest route between two points.' [20] There are, currently, multiple algorithms available such as the Bellman-Ford algorithm [4], the Dijkstra algorithm [5] and the A* Algorithm which reduces computational time significantly [18].

In the ship routing optimization field, academic research concentrates on using these algorithms along with weather forecasts for wave and wind data (i.e. Padhy [2], Szłapczyńska and Śmierzchalsk [10], Takashima [11], Wei & Zhou [19], Mannarini [6] as well as Larsson and Simonsen [16]) cover examination for large distance routes.

On the contrary, Manel Grifoll's [14] research team took a closer approach to Short Sea Shipping (SSS) optimization, with the creation of a MatLab script called SIMROUTE as a weather ship routing. There, the influence of the resolution of weather predictions on optimum routes was observed. Moreover, Lluis Basiana's dissertation [13] studied the feasibility of SIMROUTE which obtained different results depending on the wave height, wave direction, and ship's speed. Finally, more recently, Clara Borén [1] compared different parametrizations of the wave effect on navigation for weather ship routing and implemented Aertssen, Khoklov, and Bowditch's formulations.

In 2018, there was previous work by the author of this project on AIS data implementation for the International Association of Maritime Universities Student Conference (IAMUS Conference). [23] The presentation brought a first approach to the current work, carrying out different comparisons between real routes and optimum ones by manually introducing a database of routes. *(See Annex 5)*

1.2. Objectives

This project has been motivated by the need to provide tools of assessment and improvement for SIMROUTE. Until now, nearly the majority of the simulations and studies generated were from a theoretical point of view. With the new additions implemented here, a first step is made to allow users to use real-life data and be able to study the advancements on this software.

Consequently, the main objective of this End of Degree Project is to introduce and analyse real AIS data into SIMROUTE and compare it with optimum routes which have been obtained using the algorithm.

In addition, the specific objectives are:

- To create a tool to analyse AIS data using MatLab, and obtain information such as average speed, distance navigated, origin and destination.
- To create a tool to evaluate and compare optimum routes with AIS routes, by comparing their data of time and distance navigated and obtain figures with their routes.
- To provide with a short and easy guide for the use of the new modules, which is proposed to be included in SIMROUTE's manual. *(SIMROUTE's manuals are attached in Annex 4).*
- To carry out a case study in order to evaluate the introduced tools.

In this document, firstly, there is a brief explanation of the new added scripts and what their functionalities are, their complete code is written in the annex. Secondly, a step-by-step explanation is found on how to use these additions for comparison studies. Thirdly, there is a short analysis carried out to show the potential and limitations of this software, its results will be presented in tables and figures and later discussed in Chapter 4. Finally, the dissertation is concluded with recommendations for future projects and a concise evaluation of the finished work.

Chapter 2. Methods

The following chapter will cover the new additions to a new version of SIMROUTE, that will allow users to analyse routes obtained from AIS sources. Firstly, there is an introduction of current SIMROUTE and its main scripts, which have not been modified but will be used later. Secondly, there are two sections about the sources of data that will be used for the purpose of this project, AIS data and weather forecasts. Thirdly, three new tools are presented, fulfilling some specific objectives of this work. These new MatLab scripts are explained with their function, needed inputs and outputs. Fourthly, there is a proposition to update SIMROUTE's current user's manual to include a new section for comparison studies. Finally, a case study is carried out in order to evaluate the newly introduced changes.

2.1. Current SIMROUTE

SIMROUTE is a maritime route optimization software, developed by a team of researchers at UPC-BarcelonaTech, in MatLab language which performs simulations on short and long journeys, providing with optimum and minimum distance routes. It covers the Mediterranean Sea zone fully or partially with the introduction of the desired longitudes and latitudes. The current version needs as input the following parameters: port of departure, port of arrival, vessel's speed, and wave field for the days of the voyage. *For more information about SIMROUTE, please, see Annex 4.*

The main scripts used to carry out a simulation are presented next in the correct order of launching:

I. *Start.m*: it is the initializing code that links the whole SIMROUTE folder so that the next scripts can work effectively using the introduced data in the distinct folders, tools, and modules that form SIMROUTE.

II. *Make_mesh.m*: this module creates a computational mesh for the Mediterranean Sea which defines the maximum and minimum longitudes and latitudes. It is important to consider its resolution as it determines the computational time of the analysis.

III. *Make_waves.m*: with wave field forecasts for the desired route, which have to be introduced as daily forecasts for each day of the journey, it interpolates the wave height and direction for the above created mesh.

IV. *Make_sea_step1.m*, *make_sea_step2.m*, and *make_sea_step3.m*: is a process of 3 modules which allow the user to create 'sea' nodes where the wave field is considered 'land'. This deals with the problem produced by low resolution wave forecasts which can have big nodes of land where there should be sea, usually near island areas.

V. *Find_ports.m*: this small script finds the nearest node to the coordinates given by the user.

VI. *Simroute_opt.m*: this is the main module of SIMROUTE, and it calculates the optimum route between two given nodes of the mesh. The required inputs are: start node, end node, the generated *.mat* file by *make_waves.m*, vessel cruising speed, and the desired algorithm to use (Dijkstra or A*).

VII. *Simroute_Dmin.m*: on the other hand, this module calculates the minimum distance route between the start and end nodes.

VIII. *Plot_route_and_waves.m:* is a module which produces a number of figures to visualize routes and the wave field for each waypoint of the route. This allows to evaluate the influence of waves on the studied routes.

2.2. AIS as a tool for ship weather routing

The IMO defines the Automatic Identification Systems as 'those systems designed to be capable of providing information about the ship to other ships and to coastal authorities automatically'.

'Regulation 19 of SOLAS Chapter V – Carriage requirements for shipborne navigational systems and equipment, sets out navigational equipment to be carried on board ships, according to ship type. In 2000, IMO adopted a new requirement for all ships to carry automatic identification systems.'

'The regulation requires AIS to be fitted aboard all ships of 300 gross tonnage and upwards engaged on international voyages, cargo ships of 500 gross tonnage and upwards not engaged on international voyages, and all passenger ships irrespective of their size. The requirement became effective for all ships by 31 December 2004. '

The AIS is, in fact, a Very High Frequency (VHF) system that is integrated with ECDIS (Electronic Chart Display and Information System) and transmits information between ships, allowing to easily identify them and reach them via Digital Selective Calling (DSC). [9]

Returning to the IMO, the regulation requires that AIS shall provide information (including the ship's identity, type, position, course, speed, navigational status and other safety-related information) automatically to appropriately equipped shore stations, to other ships and aircraft, receive automatically such information from similarly fitted ships, monitor and track ships and exchange data with shore-based facilities. [7]

This great quantity of data is constantly being sent and it involves especially important advancements for future integrated systems of marine traffic management. From the academic point of view, it is an interesting source for the study of traffic and development of new related software. Being able to receive and process data simultaneously, approaches a superior development for real-time ship routing, including variables such as berth congestion, other ship's journeys, and other limitations.

AIS data is not only available through VHF stations, databases are another source that comes available for specific studies. For instance, the available options at the Barcelona School of Nautical Studies (FNB) are MarineTraffic [15] (accessible through FNB's login details), and a VHF station located on the attic of FNB's faculty. The station stores data for the range of 20-40 nautical miles, and the amount of data that it obtains is superior to that of an online-shared option. However, due to its small range it only covers the area of coastal Barcelona. Therefore, to cover the Mediterranean sea, an online database is more appropriate, the amount of data for a specific ship is smaller in terms of the quantity of waypoints but it has been verified that it is likewise a reliable source.

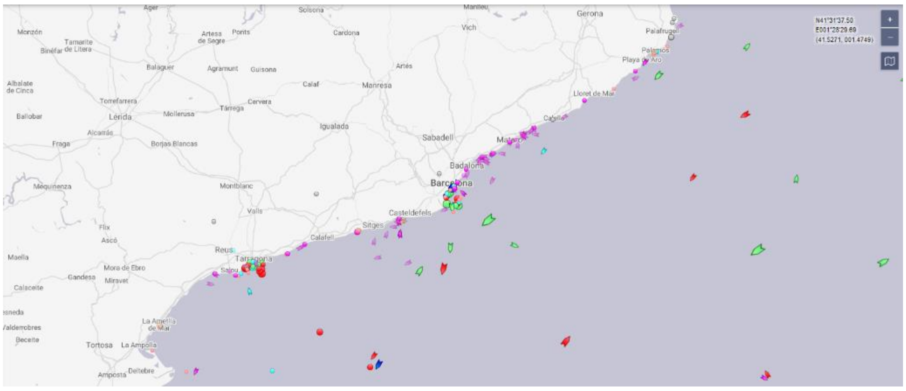

Figure 1 - Example of Marine Traffic's live map. *Source: www.marinetraffic.com*

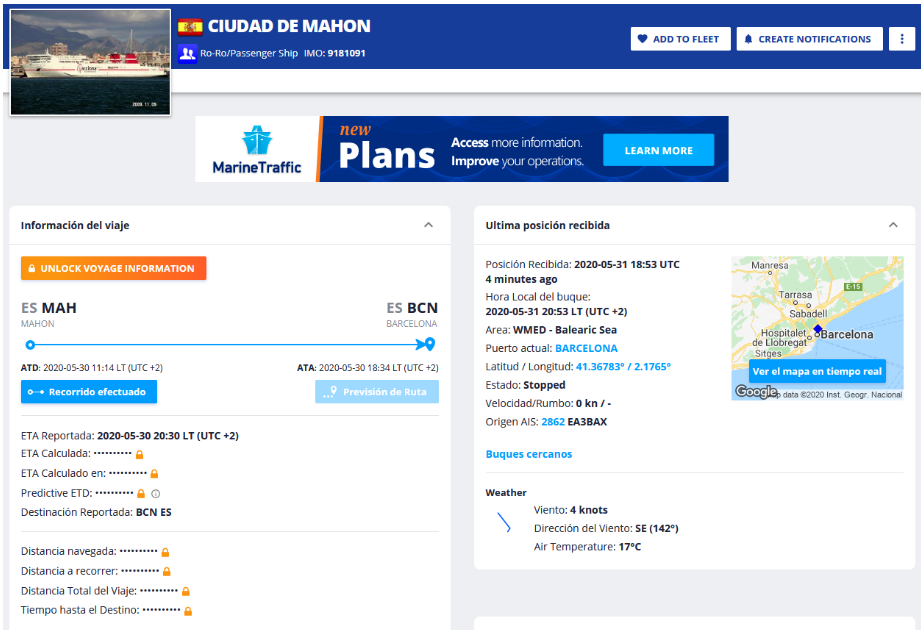

Figure 2 - Example of Marine Traffic's ship information. *Source: www.marinetraffic.com*

2.3. Weather forecasts

In order to obtain an optimum route, SIMROUTE needs some parameters, and one of them are weather forecasts. In this case, the information needed comes from wave field forecasts. They are composed of various variables of wave definition, such as wave significant height and wave direction.

SIMROUTE user's guide instructs to download wave fields in *netcdf* format from *Puertos del Estado* website. Each file poses the wave conditions hourly for one entire day, that is, if the desired route takes 3 days it will need 3 wave fields. [21]

Back in 2018, for the first attempt of AIS data introduction, 'Puertos del Estado's wave fields were provided with high resolution. However, as of 2019, big changes have been introduced into the service and high-resolution wave fields are not supported anymore. The offered forecasts now are two options: the area of the Spanish coast and western Mediterranean, or the Atlantic Ocean and Mediterranean Sea. Although it may seem a good source of data, the decrease of resolution has a major impact on SIMROUTE's optimized routes. The following are a few examples of the service offered by *Puertos del Estado. [17]*

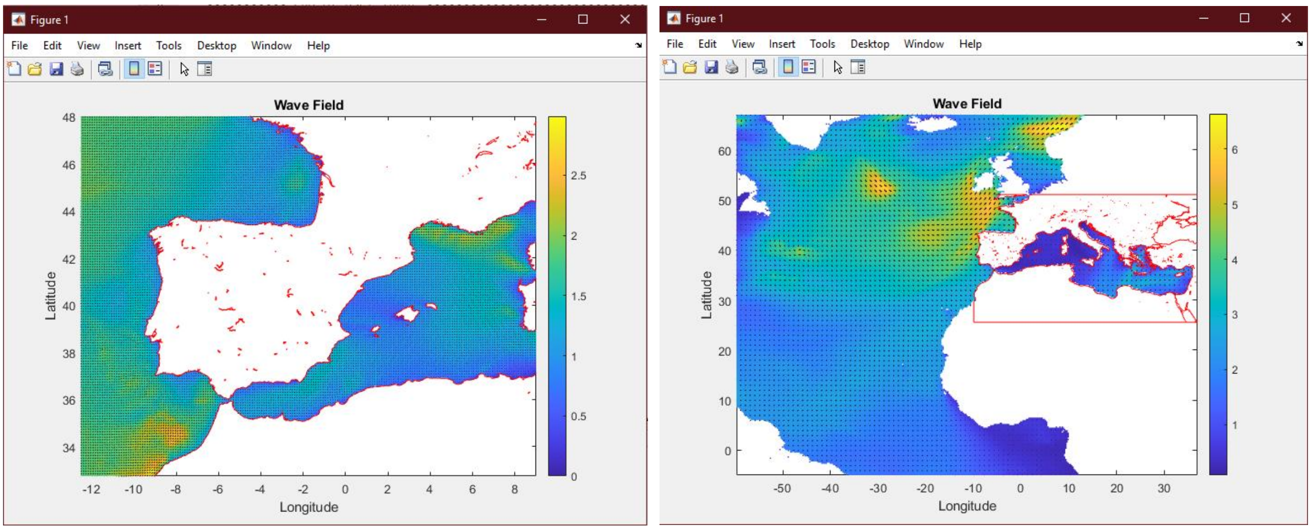

Figure 3 - Examples of Puertos del Estado wave field forecasts.

See that the first map covers the whole coast of Spain from the longitudes of -12º to 8º.

The second map covers the area of the Mediterranean and a great part of the Atlantic Ocean from latitudes 0º to 60º approximately.

Although for the objective of this work it looks suitable to use the forecasts that cover the whole of the Mediterranean Sea, the following images show issues with resolution:

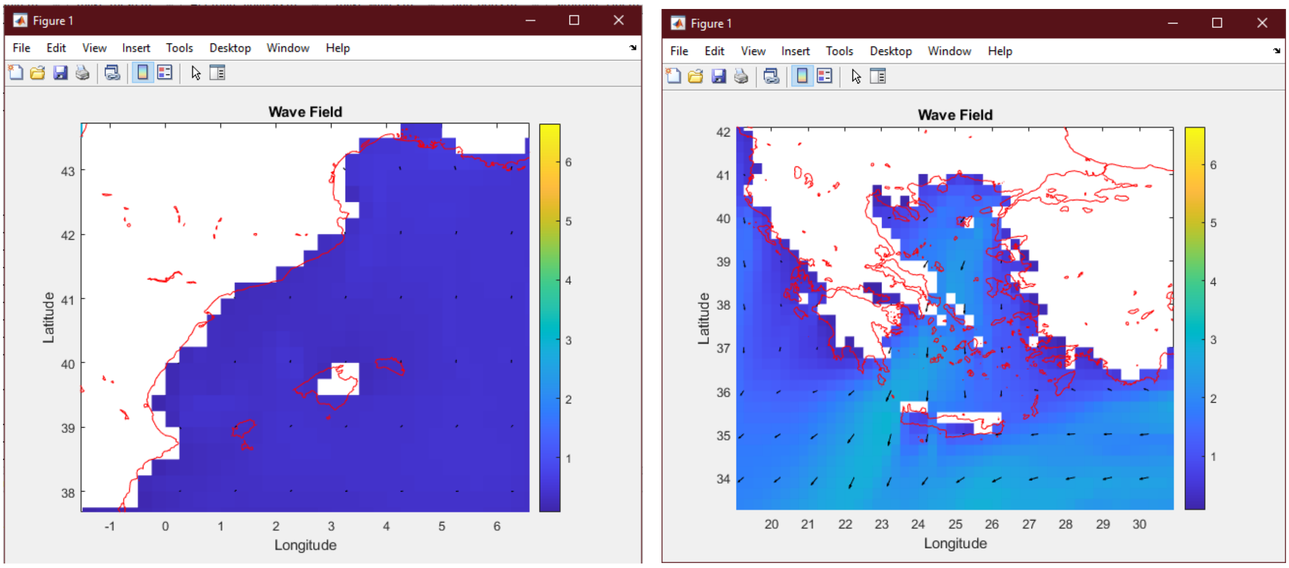

Figure 4 - Zoom in of the Atlantic Ocean and Mediterranean Sea wave forecast from Puertos del Estado.

The first image shows the area of the north-eastern Spanish coast. It can be observed that the resolution of the map does not correctly show the main ports (i.e. Valencia, Tarragona, Barcelona, Menorca, Palma de Mallorca, and Ibiza).

The second image shows the area of the Greek islands. It is noticed that most of the islands are not shown as 'land' by white blocks. On the contrary, this lack of resolution could create false routes.

Observing how this can create problems in various analysis, an alternative is searched.

Another source of wave field data is CMEMS (Copernicus Marine Environment Monitoring Service), available from May 2015. Defined by their website as 'been designed to respond to issues emerging in the environmental, business and scientific sectors. Using information from both satellite and in situ observations, it provides state-of-the-art analyses and forecasts daily, which offer an unprecedented capability to observe, understand and anticipate marine environment events.' [3]

Afterwards, there is a brief explanation on how to access it in the step-by-step guide for the use of the scripts presented in this work. To find the wave field files, one has to search in 'Ocean Products' for 'Mediterranean Sea wave analysis and forecasts'. This service allows to choose inside a designated area that covers the whole zone of the study, and to choose the dates and hours of the days wanted to download. Therefore, to download an equivalent to the aforementioned *Puertos del Estado*, one has to choose the time range of one day. Note that the Copernicus wave field forecasts have been chosen as the suitable source for this project.

Copernicus wave forecasts show better resolution. Here are some images that prove it:

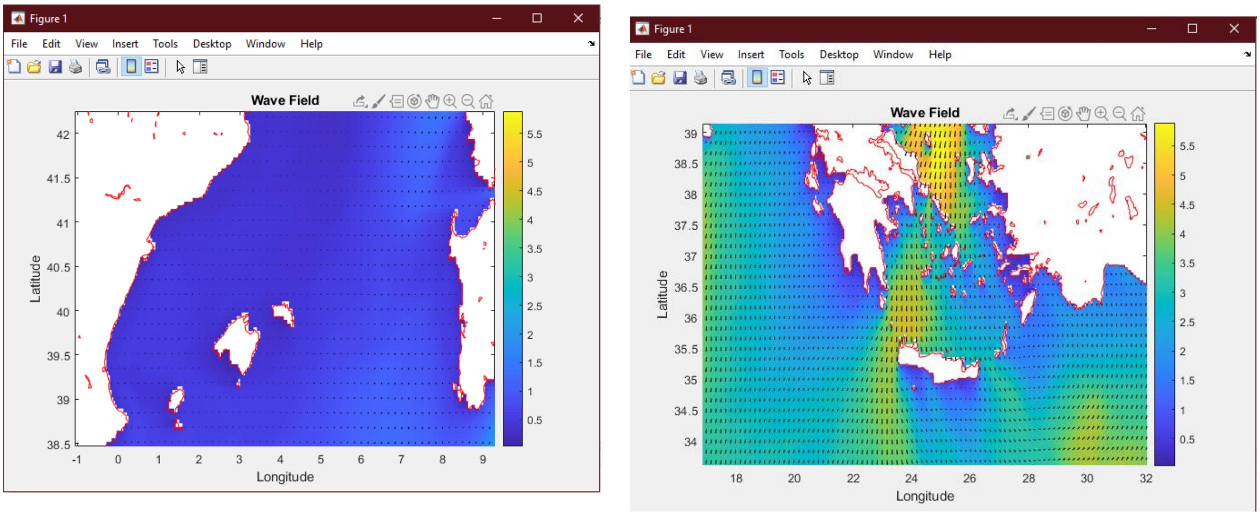

Figure 5 – Zoom in of the Mediterranean Sea wave forecast from Copernicus.

Both images show now better resolution, considering most of the islands and ports that had issues with big white blocks in the Balearic islands and the Greek islands.

2.4. New introductions to SIMROUTE

The following are three new additions to SIMROUTE which will provide it with tools to use data obtained from AIS databases and analyse it as a route. It will also be able to compare the AIS route with an optimum route created using the algorithm. In addition, SIMROUTE's *find_ports.m* module has been updated in order to introduce more functions, increase time-efficiency of SIMROUTE, and improve the interactions with the user.

2.4.1. *AIS_route_analysis.m*

This newly added module briefly presents the characteristics of a downloaded *.csv* file containing a route from AIS sources, in this case, the source of files has been MarineTraffic's database using the FNB's profile. The output will provide with information such as the time of departure and arrival, the total time and distance navigated, the average speed of the trip, and the coordinates of origin and destination. The file shall have the data presented in the following way:

a) Header: Timestamp,Source,Speed,Latitude,Longitude. (The order is not important as the script calls each variable by its name).
b) Data: date(yyyy-MM-dd HH:mm:SS), speed(xx.xx), course(xxx), latitude(xx.xxxx), longitude(xx.xxxx).

Table 1 – Example of a *.csv* file obtained from MarineTraffic. The data has been accessed via Microsoft Office's Excel program and presented here as a table. Each line of information only uses one cell.

Timestamp,Source,Speed,Course,Latitude,Longitude
2020-01-29 04:16:50,Terr-AIS,14.5,109,39.41891,-0.2781917
2020-01-29 04:21:26,Terr-AIS,16.6,76,39.42105,-0.2527583
2020-01-29 04:32:02,Terr-AIS,17.4,51,39.43914,-0.19026
2020-01-29 04:41:02,Terr-AIS,18.0,40,39.47389,-0.1534917
2020-01-29 04:50:38,Terr-AIS,17.2,43,39.50653,-0.1087083
2020-01-29 05:02:26,Terr-AIS,17.3,32,39.55163,-0.06571667
2020-01-29 05:11:40,Terr-AIS,17.3,35,39.58803,-0.03202333
2020-01-29 05:22:16,Terr-AIS,17.2,35,39.62947,0.006385
2020-01-29 05:31:16,Terr-AIS,17.3,38,39.66437,0.03966
2020-01-29 05:40:04,Terr-AIS,17.3,39,39.6972,0.07463167
2020-01-29 05:49:38,Terr-AIS,16.6,40,39.7285,0.1176633
2020-01-29 06:00:50,Terr-AIS,17.2,35,39.7729,0.15629
2020-01-29 06:10:26,Terr-AIS,17.1,35,39.80994,0.191535
2020-01-29 06:20:01,Terr-AIS,17.0,35,39.84719,0.2259383
2020-01-29 06:29:26,Terr-AIS,17.1,35,39.88382,0.2595767
2020-01-29 06:39:50,Terr-AIS,17.2,35,39.92416,0.2969767

Once the script is run, the output will be as the following example:

departure =

datetime

2020-01-29 04:16:50

arrival =

datetime

2020-01-29 11:22:25

Total time in reality, obtained from AIS data 7.0931 hours

Distance obtained from AIS data 119.0299 miles

Average speed of 16.5682 kn

Port of origin 39.4189º -0.27819º

Port of destination 41.0164º 1.185º

See Annex 1 for the complete written script of the module.

2.4.2. Updated *find_ports.m* module

As stated in section 2.1, the older *find_ports.m* module would find the nearest node to input coordinates. However, it would not tell if that node belonged to the actual study area, that is the area covered by the wave field and classified as 'sea'. That information would be obtained later when using *simroute_opt.m,* which would display an error message saying that the introduced origin and/or destination nodes were 'land', and therefore not possible to be used as the start or end of the route. Now, with the creation of this module, it is capable of finding the start and end node at the same time and it will also plot the wave field and AIS route (if desired) to allow the user to examine it before using *Simroute_opt.m*. Therefore, it translates into a great time saving tool because the earlier method would involve a few trial and errors until the correct nodes for the ports were obtained. Here is some information of the module:

The inputs for this script will be:

a) Start coordinates
b) End coordinates
c) Wave field: obtained as *.mat* from *make_waves.m*
d) AIS route: in *.csv*

And the outputs will be:

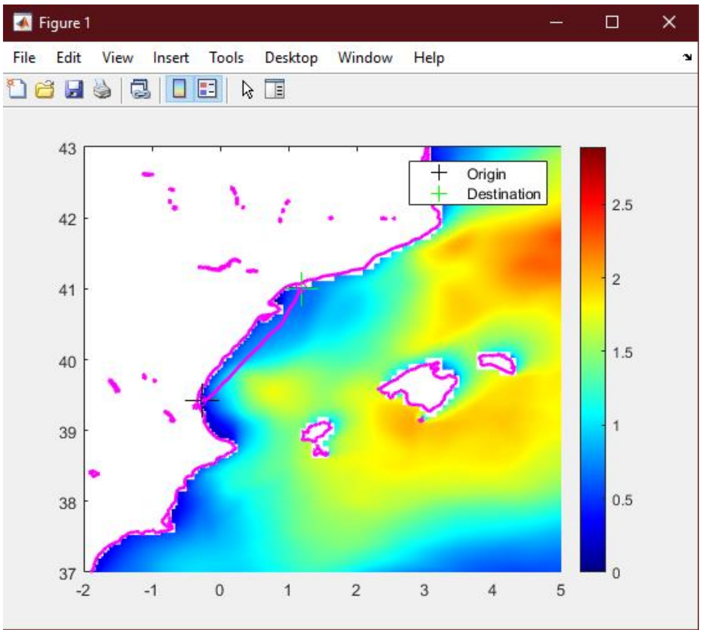

Figure 6 – Illustration of the output of *find_ports.m*

The figure shows the mask for the continent (in this case, the area of Catalonia, Valencia and Balearic Islands), the route, and the wave field to see the zone covered by it. Note that the origin and destination are marked in black and green respectively.

The port of ORIGIN coordinates are 39.42º -0.26º

node is 61150

The port of DESTINATION coordinates are 41.0164º 1.19º

node is 101232

See Annex 2 for the complete written script of the module.

2.4.3. *compare_routes.m*

This module serves as a tool to obtain a comparison between an AIS obtained route and a SIMROUTE optimum route. Its function is to collect both routes' information and process it to show useful data for a comparison. In order to do that, it obtains the data contained in the optimum route *.mat* file, and then reads the information in the AIS *.csv* file. With a few calculations it obtains a list of parameters that are displayed in the end. Mainly, it summarizes the two routes by showing the following data:

- Total time of both routes' journey.
- Total distance sailed by both journeys. (The distance sailed by the AIS route is calculated with orthodromic navigation between each waypoint, *see Annex 5 for more information)*.
- Average speed of the AIS route.
- Savings in terms of time of the SIMROUTE route.
- Savings in terms of distance of the SIMROUTE route.
- Port of origin.
- Port of destination.

The inputs for this module must be:

a) A *.csv* file containing the AIS route.
b) A *.mat* file containing the SIMROUTE route.
c) Name of the desired output figure.

The outputs will be:

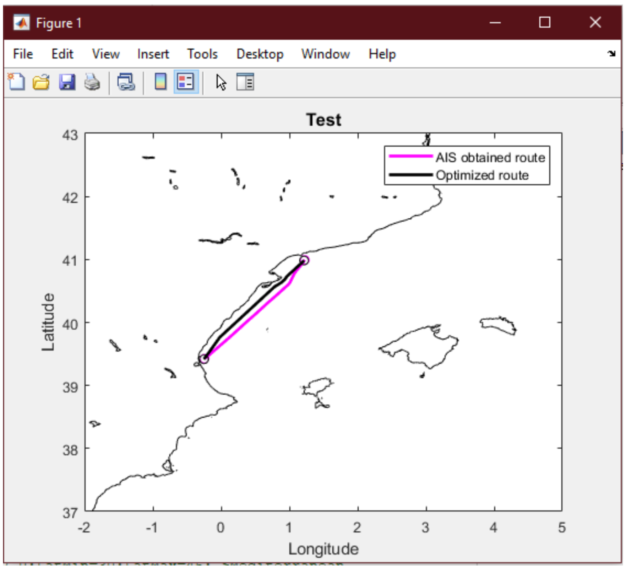

Figure 7 – Illustration of the output of *compare_routes.m*

The figure shows the mask for the continent (in this case, once again, the area of Catalonia, Valencia and Balearic Islands), the AIS route is represented in magenta and the optimal route is represented in black.

Total time with optimization	6.8353 hours
Distance with optimization	116.0161 miles
Total time in reality, obtained from AIS data	10.0667 hours
Distance obtained from AIS data	116.4468 miles
Average speed of	11.502 kn
Saved	3.2314 hours
Saved	32.0997 % (h)
Saved	0.43075 miles
Saved	0.36991 % (miles)
Port of origin	39.425º -0.244º
Port of destination	40.993º 1.2268º

See Annex 3 for the complete script code.

2.5. Update of SIMROUTE's User's manual

SIMROUTE incorporates two short guides to allow new researchers take part in the on-going project.

Firstly, the SIMROUTE TECHNICAL MANUAL instructs one on the theory that is behind the code; from the basis and algorithm of an optimum pathfinder, wave interaction and parametrization for its incidence to the ship, useful tips for the use of the scripts, and theory on the emissions of pollutants obtained from the script *make_emissions.m.*

Secondly, the SIMROUTE USER'S MANUAL is a short course on SIMROUTE, with an explanation on what it is, how it is placed in the educational framework, its different scripts and their functions, and a step-by-step guide with some examples.

(See Annex 4 for the original SIMROUTE manuals)

The proposition of this project is to introduce an update of SIMROUTE's User's Manual with a step-by-step guide on the use of SIMROUTE for AIS route analysis and comparison, it also includes an update of the use of the new updated *find_ports.m* module. And finally, there is a flowchart to summarize and simplify the explanation.

2.5.1. Analysis and comparison for AIS routes' guide proposition

The following is the new proposed step-by-step guide for analysis and comparison of AIS routes:

Using SIMROUTE's AIS route analysis functionality step-by-step:

1. Download from www.marinetraffic.com a *.csv* file containing the desired route to analyze. Remember it must have the following data: date and time, speed, course, latitude, and longitude.

2. Open MATLAB. Open *start.m* inside SIMROUTE's folder and run it.

3. Open *make_mesh.m*, choose mesh dimension in degrees; define *LonMin*, *LonMax*, *LatMin* and *LatMax* which stand for minimum and maximum longitude and minimum and maximum latitude. Next, choose mesh resolution in minutes. Run it.

4. Open *AIS_route_analysis.m*, write the name of the *.csv* file on the variable *filename*. Run it. This will present the main data for the AIS route which will define the SIMROUTE optimum route. Take notes of the time and dates of departure and arrival, these will be the days for the wave field forecasts. Also, the time of departure will define the *Tini_trip* (Initial time variable) of *make_waves.m*. There is also the coordinates of the origin and destination and some other useful information such as the total time of the route, the total distance, and the average speed.

5. Download from https://marine.copernicus.eu/ the desired wave field. You will need to sign up if you do not have an account, it is free for research. The wave fields can be found on Ocean products (DATA) under the name of 'Mediterranean sea waves analysis and forecast'. Once in there, choose 'MED00-HMCR-WAV-AN-FC-H' from the list, you will, then, have to set the desired

geographical area and the dates for which you need the forecast. Make sure to download one-day forecasts as *make_waves.m* will only accept those. Therefore, you need to set start date for a given date at 00:00 and end date for the same date at 23:00. Download it.

6. Open *make_waves.m*, set *Tini_trip* to the time given on departure before. Type the name of the wave field files in the *ARX* variable and name the output file on *arxiu_out*. Run *make_waves.m*. Depending on the resolution and the amount of days introduced it may take some time. Once finished it will display 'Saving the swell' and the time consumed by the script.

7. Open *find_ports.m*. Now that the wave field is done you need to find out the nodes for start and end. The main use of this script is to find the nodes by loading the wave field and writing the start coordinates on *Lon_port1* and *Lat_port1* (longitude and latitude) and then the end coordinates on *Lon_port2* and *Lat_port2*. The coordinates must be given in degrees. There is also the option to display the wave field and/or the AIS route to be able to see how the chosen coordinates compare to the real route that is going to be analysed. You can do that by changing the variables *WF* and *AISR* values to '1' (make sure to load the AIS route in case you want to plot it). Run the script and take notes of the ORIGIN and DESTINATION nodes. If the nodes happen to be out of the wave field, the script will display that the given node 'is land'.

8. Open *simroute_Opt.m*. Write the *make_waves.m* wave field file name. Introduce the start and end nodes in the variables *nod_ini* (start) and *nod_end* (end). Choose the name for the output file and choose the cruising speed of the vessel in knots. Check that everything has been introduced correctly and run the script. This one may take some time, for Western to Eastern Mediterranean routes it may take up to 1h30m. Once it has finished it will display some information of the created optimum route.

9. Open *compare_routes.m*. Type the desired name for the output figure. Write the AIS route file name in the variable *filename* and the SIMROUTE optimum route in the variable *q*. Choose the desired dimension of the figure by setting the maximum and minimum latitude and longitude in *LonMin*, *LonMax*, *LatMin*, and *LatMax*. Run the script. Now, you will obtain information about both routes and a saved figure in the 'out/compare_routes_figures' folder of SIMROUTE.

2.5.2. Analysis and comparison for AIS routes' flow chart proposition

To carry out a comparison between two routes, here is the following process diagram:

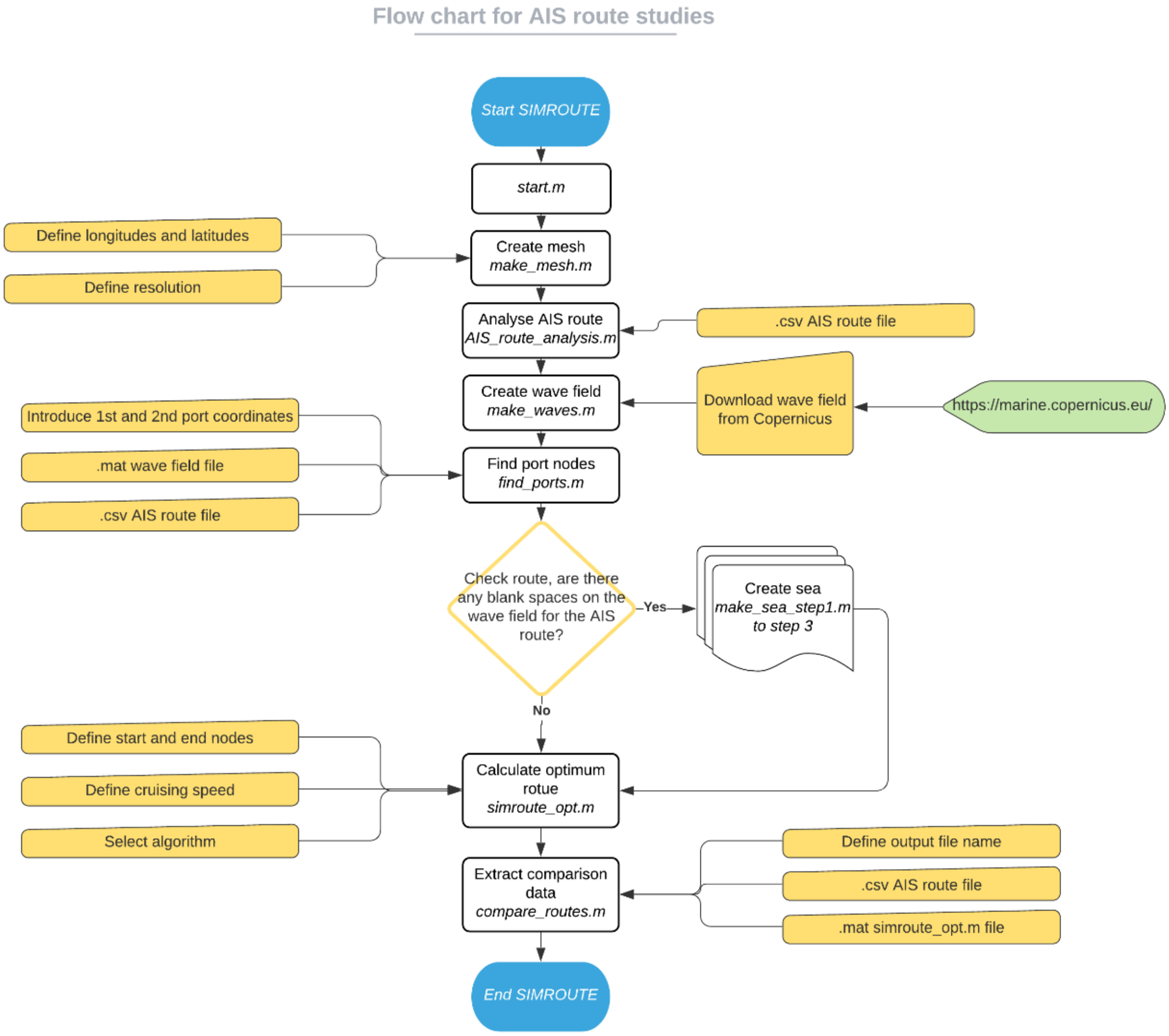

Figure 8 – Flow chart for the use of the new SIMROUTE for comparison studies.

2.6. Case study

Following the addition of the new modules, a case study is carried out using the described procedure. For this purpose, 4 cases have been evaluated, with 4 equivalent ships with the same characteristics and routes that cover the whole Mediterranean and navigate through areas that can be challenging with bad resolution wave maps (that is areas such as the Greek islands or the Messina Strait). The reason why these cases have been chosen is that they are a good opportunity to observe and evaluate the inconveniences that may appear when using this software to compare real cases with optimum routes, large distance routes are preferred to increase the influence of any flaws and also have the opportunity to observe more energetic wave episodes. The data for the ships of this route has been obtained from MarineTraffic by a fellow student that is doing research on their routes, and that has asked to keep their names confidential. Coprenicus wave field forecasts are used for this study. The following tables show the main data of the examples.

Table 2 - Ship defining data, including: speed (in knots), length (in meters) and deadweight DWT (in tons).

Cruise speed	17	kn
Length	260	m
DWT	50530	ton

Table 3 – Route taken by the ships in the analysis.

ROUTE	
1. Valencia (Spain) - ESVLC	6. Aliaga (Turkey) - TRAGA
2. Tarragona (Spain) - ESTRG	7. Piraeus (Greece) - GRPIE
3. Mersin (Turkey) - TRMER	8. Livorno (Italy) - ITLVN
4. Ashdod (Israel) - ILASH	9. Barcelona (Spain) - ESBCN
5. Haifa (Israel) - ILHFA	10. Valencia (Spain) - ESVLC

Table 4 – Dates of the trips for each case.

Case 1	**Case 2**	**Case 3**	**Case 4**
29/01/2020-18/02/2020	30/12/2019-25/01/2020	24/01/2020-11/02/2020	05/01/2020-28/01/2020

The chosen speed for the analysis will be the average speed of the AIS route for the cases 1 and 2, and the cruise speed for cases 3 and 4.

Chapter 3. Results

Here are the results for the analysis carried out on Cases 1 to 4.

3.1. Case 1 29/01/2020 to 18/02/2020

Table 5 – General table of the results for Case 1 analysis (3 tables). The chosen ship speed is the average speed in knots for each trip. The % of savings in terms of time vary depending on the average speed observed in the AIS route.

	ESVLC - ESTRG	**ESTRG-TRMER**	**TRMER - ILASH**
DATE	29/1/2020	29/1/2020 - 02/02/2020	03/02/2020-04/02/2020
Start Lat	*39.42*	*40.99*	*36.67*
Start Lon	*-0.28*	*1.44*	*34.67*
Start node	**61149**	**1681526**	**1022811**
End Lat	*41.02*	*36.65*	*31.87*
End Long	*1.19*	*34.65*	*34.56*
End node	**101653**	**1020259**	**288116**
Speed (kn)	17	17	17
Average speed AIS (kn)	16.78	18.75	12.67
Aprox speed	16.7	18.8	12.7
SIMROUTE (h)	7.04	89.42	22.78
AIS (h)	7.09	88.52	24.38
SIMROUTE (miles)	117.54	1663.78	288.75
AIS (miles)	119.03	1659.65	308.97
Savings (miles)	1.49	-4.14	20.22
Savings (% miles)	1.25	-0.25	6.54
Savings (h)	0.06	-0.90	1.60
Savings (% time)	0.80	-1.01	6.54

	ILASH - ILHFA	**ILHFA - TRAGA**	**TRAGA - GRPIE**
DATE	5/2/2020	07/02/2020-09/02/2020	09/02/2020-10/02/2020
Start Lat	*31.89*	*32.89*	*38.78*
Start Lon	*34.55*	*34.99*	*26.8*
Start node	**290667**	**443753**	**1346316**
End Lat	*32.87*	*38.78*	*37.71*
End Long	*35*	*26.64*	*23.75*
End node	**441203**	**1343755**	**1180318**
Speed (kn)	17	17	17
Average speed AIS (kn)	13.05	17.2	17.4
Aprox speed	13.1	17.2	17.4
SIMROUTE (h)	5.03	38.77	9.68
AIS (h)	5.22	38.29	9.74
SIMROUTE (miles)	65.60	642.65	168.10

AIS (miles)	71.77	648.09	169.36
Savings (miles)	6.17	5.44	1.26
Savings (% miles)	8.60	0.84	0.75
Savings (h)	0.19	-0.48	0.06
Savings (% time)	3.65	-1.25	0.66

	GRPIE - ITLVN	**ITLVN - ESBCN**	**ESBCN - ESVLC**
DATE	11/02/2020-14/02/2020	15/02/2020-17/02/2020	17/02/2020-18/02/2020
Start Lat	*37.72*	*43.51*	*41.27*
Start Lon	*23.68*	*10.2*	*2.17*
Start node	**1182864**	**2067253**	**108027**
End Lat	*43.35*	*41.29*	*39.56*
End Long	*10.19*	*2.28*	*-0.03*
End node	**2044293**	**1727494**	**64532**
Speed (kn)	17	17	17
Average speed AIS (kn)	13.3	13.7	12.4
Aprox speed	13.3	13.7	12.4
SIMROUTE (h)	67.69	27.93	11.69
AIS (h)	68.31	27.67	11.86
SIMROUTE (miles)	891.00	380.18	144.50
AIS (miles)	893.82	377.42	143.91
Savings (miles)	2.82	-2.77	-0.59
Savings (% miles)	0.32	-0.73	-0.41
Savings (h)	0.63	-0.27	0.17
Savings (% time)	0.92	-0.97	1.43

Table 6 – Summary table for Case 1. The total saved time for this round trip is 1 hour approximately, which makes up a 0.38% of the total 281.08 hours of the AIS route, very similar, to the savings of 0.68% of miles.

CASE 1	**TOTAL**
SIMROUTE (h)	280.03
AIS (h)	281.08
SIMROUTE (miles)	4362.12
AIS (miles)	4392.01
Savings (miles)	29.90
Savings (% miles)	0.68
Savings (h)	1.06
Savings (% time)	0.38

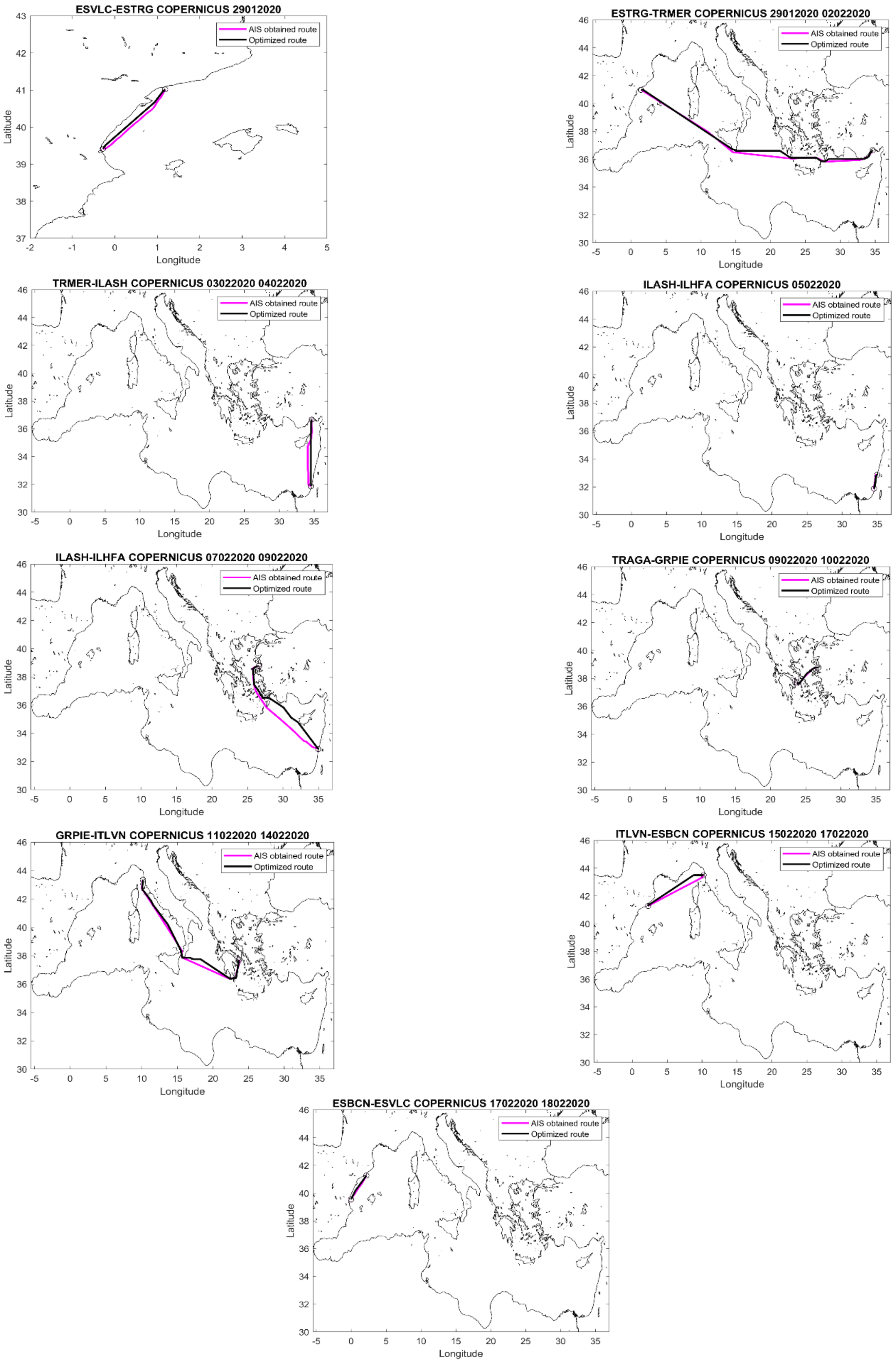

Figure 9 – Case 1 figure compilation of trips. The order is from left to right starting from the upper left corner. By reading the tables shown above, the maximum saving percentage of time has been in the Mersin (TRMER) to Ashdod (ILASH) trip, where there has been a saving of 6.54%. On the contrary, the least efficient route was Haifa (ILHFA) to Aliaga (TRAGA) with a -1.25%.

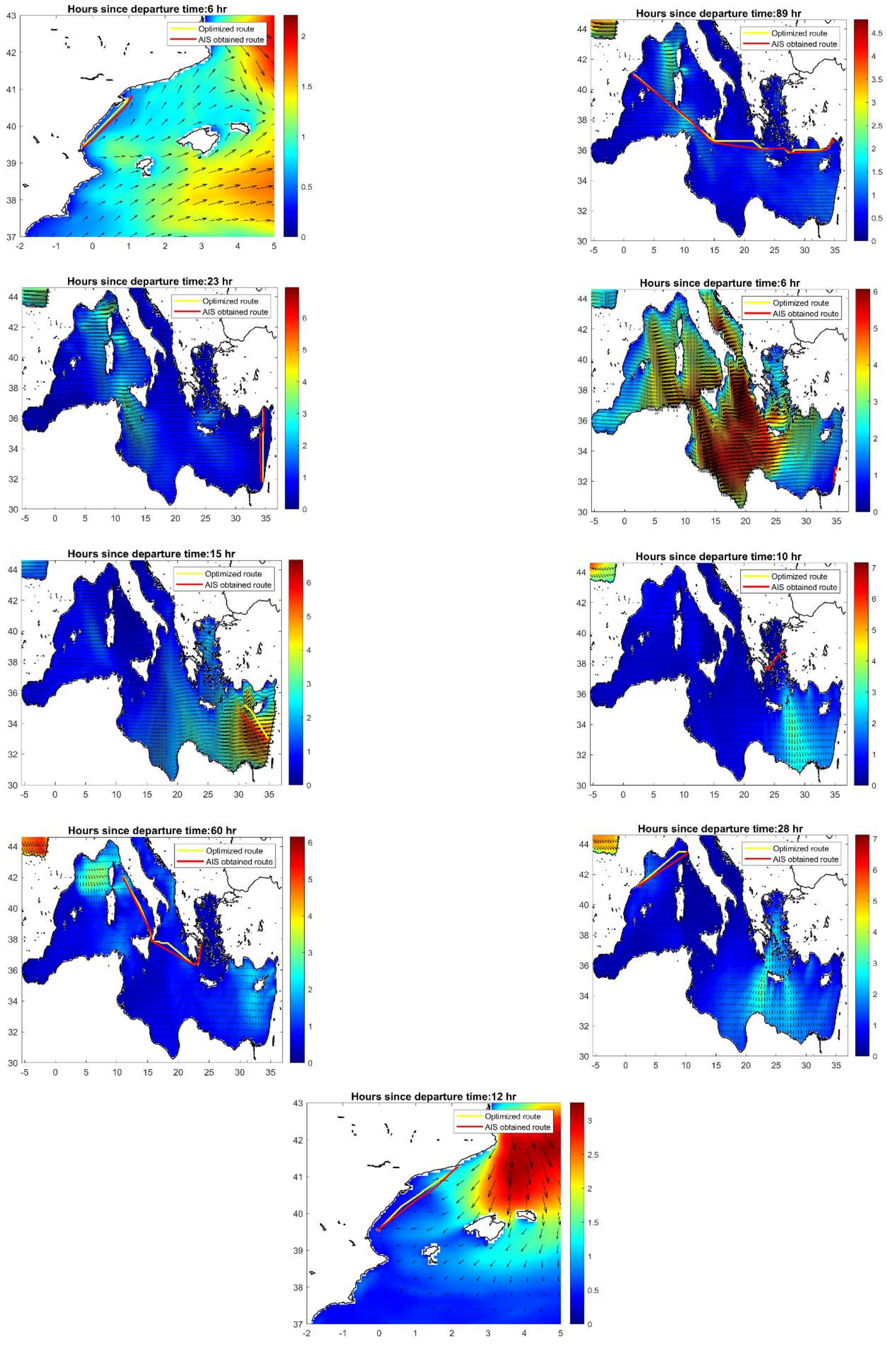

Figure 10 – Case 1 figure compilation of wave maps for the most relevant part of each trip. In this case, there are two energetic wave episodes that influence the routes, the 5th figure (ILASH-ILHFA) and the 7th (GRPIE-ITLVN). Source of weather forecasts: COPERNICUS database.

3.2. Case 2 30/12/2020 to 25/01/2020

Table 7 – General table of the results for Case 2 analysis (3 tables). The chosen trip ship speed is the average speed in knots for each trip. The % of savings in terms of time varies depending on the average speed observed in the AIS route.

	ESVLC - ESTRG	**ESTRG-TRMER**	**TRMER - ILASH**
DATE	30/12/2019	01/01/2020-05/01/2020	05/01/2020-07/01/2020
Start Lat	*39.55*	*37.41*	*36.45*
Start Lon	*-0.11*	*12.59*	*34.68*
Start node	**64106**	**1133730**	**989648**
End Lat	*40.97*	*36.51*	*32.2*
End Long	*1.15*	*34.44*	*34.21*
End node	**100388**	**997287**	**339115**
Speed (kn)	17	17	17
Average speed AIS (kn)	12.5	13.4	11.5
Aprox speed	12.5	13.4	11.5
SIMROUTE (h)	8.35	83.78	23.58
AIS (h)	7.85	81.49	23.49
SIMROUTE (miles)	104.21	1087.99	258.23
AIS (miles)	104.08	1085.00	256.27
Savings (miles)	-0.13	-2.99	-1.95
Savings (% miles)	-0.13	-0.28	-0.76
Savings (h)	-0.50	-2.30	-0.09
Savings (%)	-6.30	-2.82	-0.38

	ILASH - ILHFA	**ILHFA - TRAGA**	**TRAGA - GRPIE**
DATE	10/1/2020	12/01/2020-14/01/2020	15/1/2020
Start Lat	*31.89*	*32.89*	*38.81*
Start Lon	*34.54*	*34.95*	*26.83*
Start node	**290666**	**443751**	**1348868**
End Lat	*32.88*	*38.78*	*37.79*
End Long	*34.92*	*26.68*	*23.68*
End node	**441198**	**1343757**	**1193068**
Speed (kn)	17	17	17
Average speed AIS (kn)	12.9	19	16.1
Aprox speed	12.9	19	16.1
SIMROUTE (h)	4.97	33.89	11.07
AIS (h)	4.99	34.90	11.09
SIMROUTE (miles)	62.76	642.37	175.06
AIS (miles)	64.54	647.90	176.77
Savings (miles)	1.77	5.54	1.71
Savings (% miles)	2.75	0.85	0.97
Savings (h)	0.02	1.01	0.02
Savings (%)	0.31	2.90	0.15

	GRPIE - ITLVN	ITLVN - ESBCN	ESBCN - ESVLC
DATE	16/01/2020-19/01/2020	19/01/2020-22/01/2020	24/01/2020-25/01/2020
Start Lat	*37.75*	*43.49*	*41.23*
Start Lon	*23.66*	*10.19*	*2.13*
Start node	**1187965**	**2064701**	**1717281**
End Lat	*43.3*	*41.16*	*39.42*
End Long	*10.26*	*2.37*	*-0.19*
End node	**2034093**	**1707092**	**1441634**
Speed (kn)	17	17	17
Average speed AIS (kn)	17	15.5	10
Aprox speed	17	15.5	10
SIMROUTE (h)	52.07	39.58	15.40
AIS (h)	52.92	26.69	15.81
SIMROUTE (miles)	873.30	430.34	152.19
AIS (miles)	884.08	375.84	151.92
Savings (miles)	10.78	-54.50	-0.27
Savings (% miles)	1.22	-14.50	-0.18
Savings (h)	0.85	-12.90	0.41
Savings (%)	1.61	-48.32	2.26

Table 8 - Summary table for Case 2. The total saved time for this round trip is -13.48, which makes up a -5.20% of the 3746.40 hours of the AIS route. On the contrary, in terms of distance savings, it performed better but still it did not surpass the AIS route with a -1.07%.

CASE 2	TOTAL
SIMROUTE (h)	272.69
AIS (h)	259.22
SIMROUTE (miles)	3786.44
AIS (miles)	3746.40
Savings (miles)	-40.04
Savings (% miles)	-1.07
Savings (h)	-13.48
Savings (% time)	-5.20

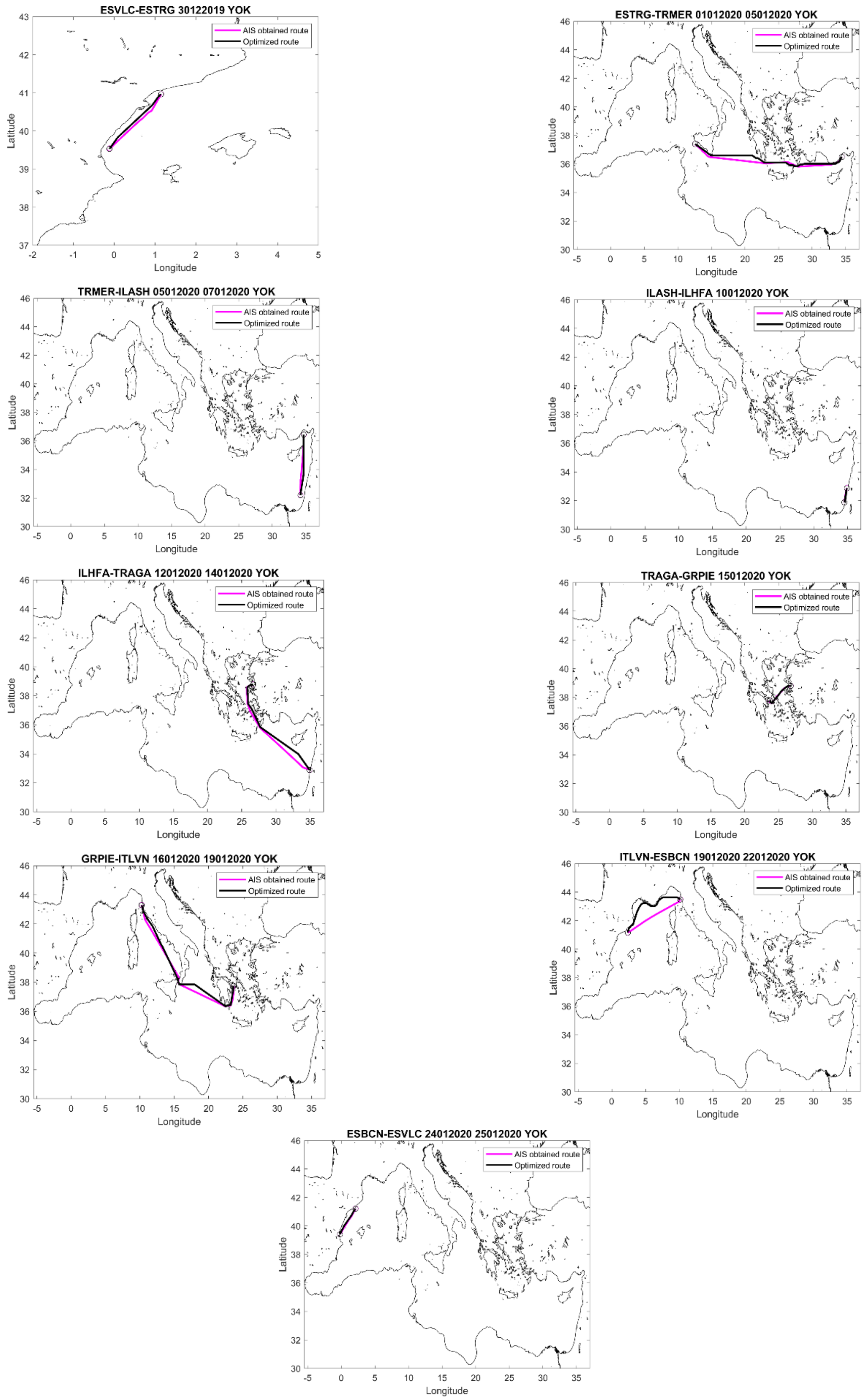

Figure 11 – Case 2 figure compilation of trips. The order is from left to right starting from the upper left corner. In the tables, it is observed that the maximum saving percentage is in the Haifa (ILHFA) to Aliaga (TRAGA) trip, with a 2.90%. However, there is a very big difference of 12.90 hours in the Livorno (ITLVN) to Barcelona (ESBCN) trip, reducing savings to -48.32%. Note that the Tarragona (ESTRG) to Mersin (TRMER) trip only had data available from a starting point near Sicily.

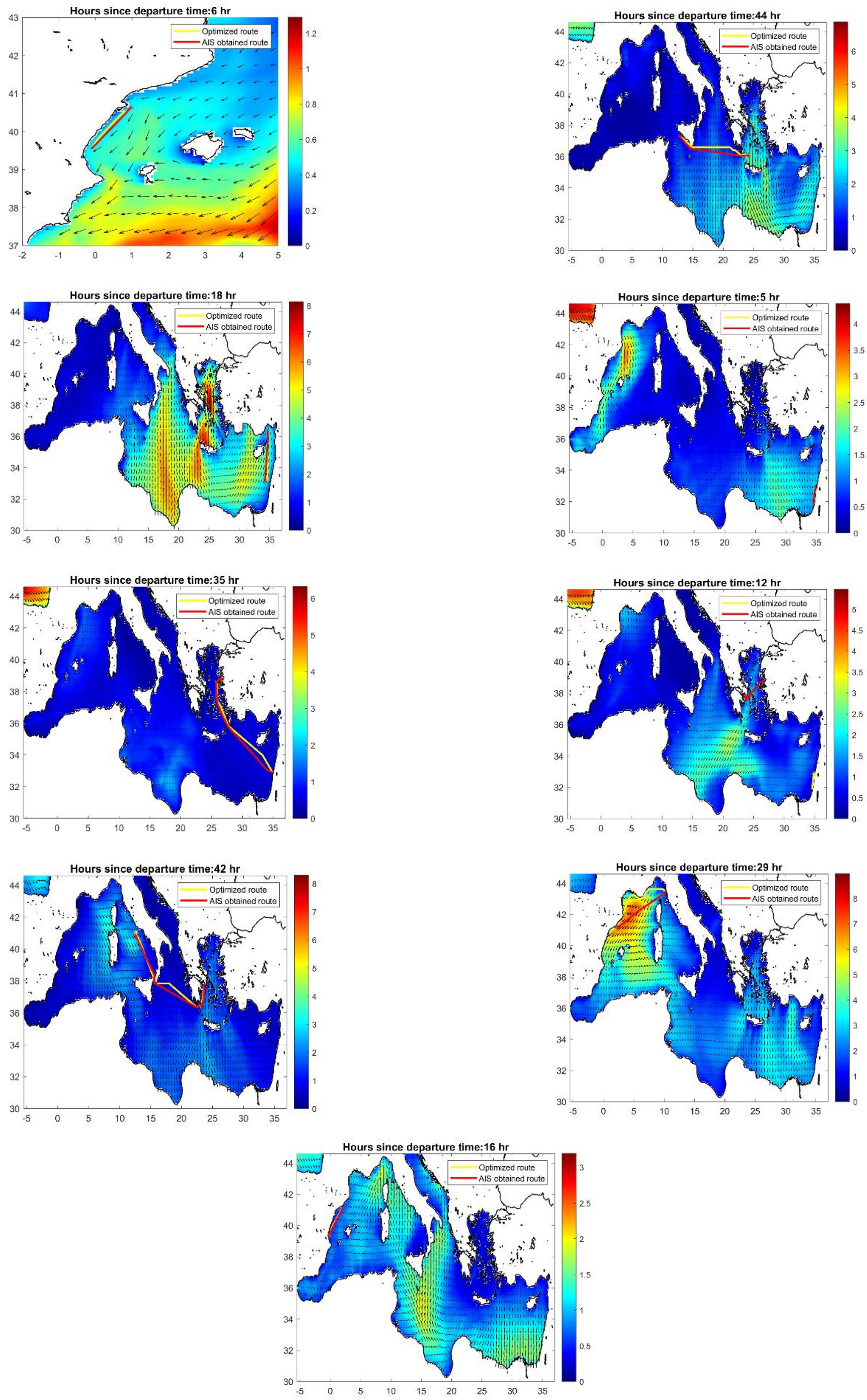

Figure 12 – Case 2 figure compilation of wave maps for the most relevant part of each trip. In this case it is observed head seas in the first figure (ESVLC-ESTRG), beam seas halfway to the destination in the second one (ESTRG-TRMER), and an energetic wave episode of following seas in the eighth figure (ITLVN-ESBCN). Source of weather forecasts: COPERNICUS database.

3.3. Case 3 24/01/2020 to 11/02/2020

Table 9 – General table of results for Case 3 analysis (3 tables). The chosen trip speed is 17 knots. The AIS routes for each trip have very different speeds from 22.2 knots to 11.5, producing big differences in savings of time. Note that the ship did not do the Ashdod to Haifa trip and, thus, there is no information in the corresponding fields.

	ESVLC - ESTRG	**ESTRG-TRMER**	**TRMER – ILHFA**
DATE	24/1/2020	25/01/2020-28/01/2020	29/1/2020
Start Lat	*39.58*	*40.86*	*36.57*
Start Lon	*-0.05*	*1.98*	*34.69*
Start node	**64952**	**1661150**	**1007506**
End Lat	*40.92*	*35.92*	*32.89*
End Long	*1.15*	*33.06*	*34.95*
End node	**99125**	**907919**	**443751**
Speed (kn)	17	17	17
Average speed AIS (kn)	22.2	21.3	17.3
Aprox speed	22.2	21.3	17.3
SIMROUTE (h)	5.80	92.19	13.23
AIS (h)	4.47	72.81	14.51
SIMROUTE (miles)	98.23	1546.90	223.52
AIS (miles)	99.51	1548.06	243.31
Savings (miles)	1.28	1.16	19.79
Savings (% miles)	1.29	0.07	8.13
Savings (h)	-1.33	-19.37	1.28
Savings (%)	-29.80	-26.61	8.83

	ILASH - ILHFA	**ILHFA - TRAGA**	**TRAGA - GRPIE**
DATE		31/01/2020-02/02/2020	3/2/2020
Start Lat		*32.89*	*38.79*
Start Lon		*34.96*	*26.79*
Start node		**443751**	**1346315**
End Lat		*38.78*	*37.84*
End Long		*26.65*	*23.66*
End node		**1343756**	**1200720**
Speed (kn)		17	17
Average speed AIS (kn)		18.3	14.2
Aprox speed		18.3	14.2
SIMROUTE (h)		38.43	10.44
AIS (h)		35.12	11.95
SIMROUTE (miles)		642.93	177.08
AIS (miles)		644.68	176.49
Savings (miles)		1.75	-0.59
Savings (% miles)		0.27	-0.33
Savings (h)		-3.31	1.52
Savings (%)		-9.44	12.70

	GRPIE - ITLVN	ITLVN - ESBCN	ESBCN - ESVLC
DATE	04/02/2020-07/02/2020	08/02/2020-10/02/2020	10/02/2020-11/02/2020
Start Lat	*37.65*	*43.53*	*41.22*
Start Lon	*23.71*	*10.24*	*2.13*
Start node	**1170111**	**2069806**	**106761**
End Lat	*43.43*	*41.27*	*39.42*
End Long	*10.19*	*2.25*	*-0.21*
End node	**2054497**	**1724942**	**61153**
Speed (kn)	17	17	17
Average speed AIS (kn)	13.9	11.5	12.9
Aprox speed	13.9	11.5	12.9
SIMROUTE (h)	54.30	22.67	8.99
AIS (h)	67.33	33.33	12.54
SIMROUTE (miles)	890.20	382.85	152.54
AIS (miles)	905.62	383.44	159.95
Savings (miles)	15.42	0.59	7.41
Savings (% miles)	1.70	0.15	4.63
Savings (h)	13.03	10.66	3.55
Savings (%)	19.35	31.97	28.28

Table 10 – Summary table for Case 3. The total saved time for this round trip is 6 hours approximately, making up a 2.39% of savings in terms of time.

CASE 3	TOTAL
SIMROUTE (h)	246.05
AIS (h)	252.06
SIMROUTE (miles)	4114.25
AIS (miles)	4161.06
Savings (miles)	46.81
Savings (% miles)	1.12
Savings (h)	6.03
Savings (% time)	2.39

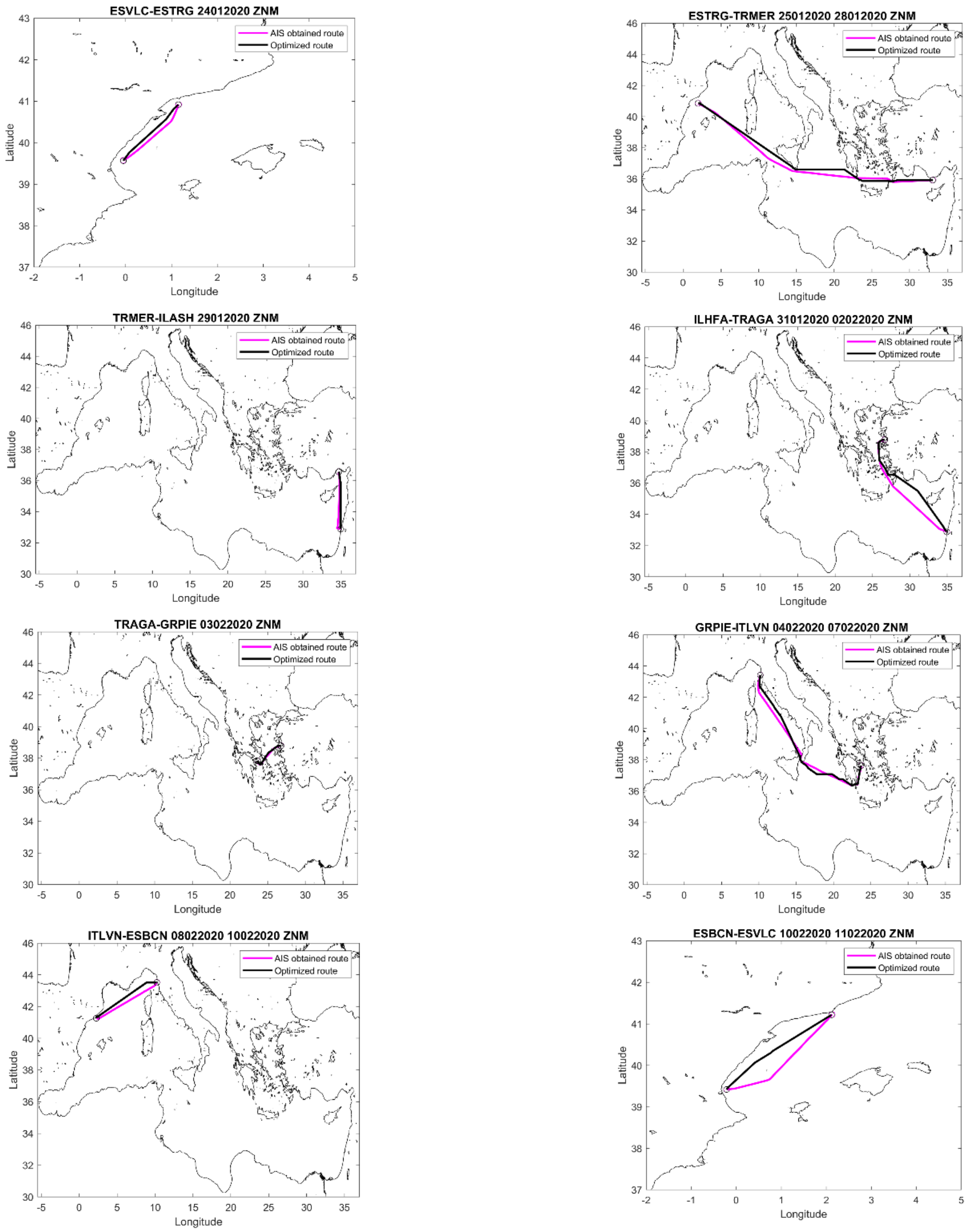

Figure 13 – Case 3 figure compilation of trips. The order is from left to right starting from the upper left corner. The maximum savings in terms of time have been in the Livorno (ITLVN) to Barcelona (ESBCN) trip with 31.97 %. On the contrary, the least efficient trip was from Valencia (ESVLC) to Tarragona (ESTRG) with a difference of -1.3 hours.

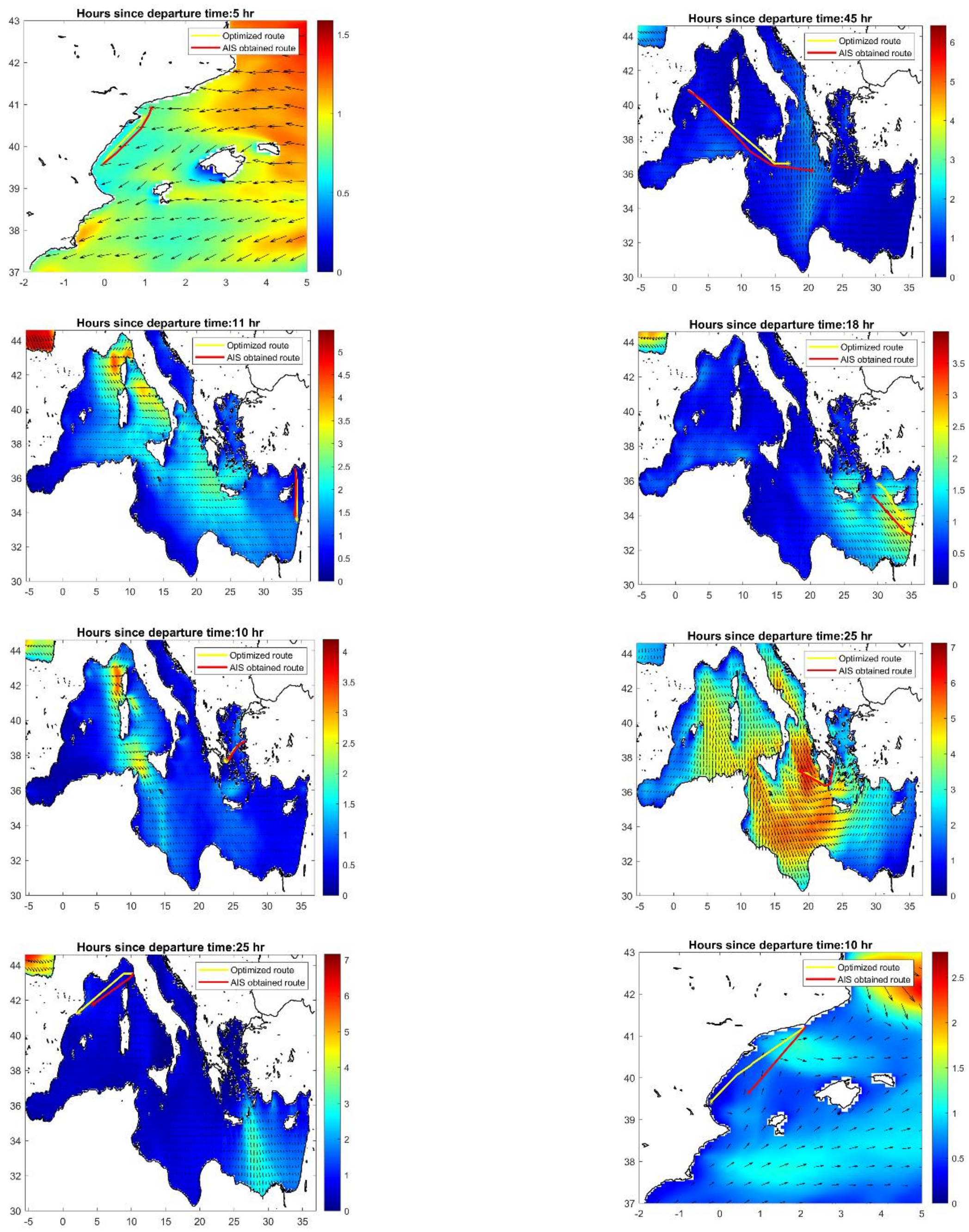

Figure 14 – Case 3 compilation of figures of wave maps for the most relevant part of each trip. In this case the most relevant wave episodes occur in the fourth figure (ILHFA-TRAGA), and the sixth figure (GRPIE-ITLVN). Source of weather forecasts: COPERNICUS database.

3.4. Case 4 05/01/2020 to 28/01/2020

Table 111 – General table of results for Case 4 analysis (3 tables). The chosen trip speed is 17 knots. There is a similar situation to that of the previous Case 3, the AIS speeds vary greatly in different trips, with a range from 10.9 knots to 20.4 knots.

	ESVLC - ESTRG	**ESTRG-TRMER**	**TRMER - ILASH**
DATE	05/01/2020-06/01/2020	07/01/2020-12/01/2020	13/01/2020-14/01/2020
Start Lat	*39.43*	*40.95*	*36.67*
Start Lon	*-0.24*	*1.59*	*34.66*
Start node	**61151**	**1676433**	**1022810**
End Lat	*40.99*	*36.49*	*31.87*
End Long	*1.23*	*34.46*	*34.56*
End node	**100813**	**994737**	**288116**
Speed (kn)	17.0	17.0	17.0
Average speed AIS (kn)	11.5	14.6	13.2
Aprox speed	11.5	14.6	13.2
SIMROUTE (h)	6.84	97.85	17.02
AIS (h)	10.07	114.04	21.67
SIMROUTE (miles)	116.02	1642.73	288.64
AIS (miles)	116.45	1642.25	300.85
Savings (miles)	0.43	-0.48	12.21
Savings (% miles)	0.37	-0.03	4.06
Savings (h)	3.23	16.20	4.64
Savings (%)	32.10	14.20	21.42

	ILASH - ILHFA	**ILHFA - TRAGA**	**TRAGA - GRPIE**
DATE	15/1/2020	17/01/2020-19/01/2020	20/1/2020
Start Lat	*31.87*	*32.97*	*38.86*
Start Lon	*34.57*	*34.63*	*26.75*
Start node	**288117**	**456486**	**1356517**
End Lat	*32.92*	*38.76*	*37.75*
End Long	*34.97*	*26.41*	*23.71*
End node	**448854**	**1341190**	**1187968**
Speed (kn)	17.0	17.0	17.0
Average speed AIS (kn)	10.9	16.6	20.4
Aprox speed	10.9	16.6	20.4
SIMROUTE (h)	3.95	36.56	10.41
AIS (h)	6.52	36.51	8.28
SIMROUTE (miles)	67.14	613.83	170.29
AIS (miles)	72.07	615.06	169.36
Savings (miles)	4.93	1.23	-0.93
Savings (% miles)	6.84	0.20	-0.55
Savings (h)	2.57	-0.05	-2.13
Savings (%)	39.41	-0.13	-25.71

	GRPIE - ITLVN	**ITLVN - ESBCN**	**ESBCN - ESVLC**
DATE	21/02/2020-24/01/2020	25/1/2020-26/01/2020	27/01/2020-28/01/2020
Start Lat	*37.73*	*43.52*	*41.24*
Start Lon	*23.66*	*10.24*	*2.13*
Start node	**1182863**	**2069806**	**107182**
End Lat	*43.32*	*41.33*	*39.42*
End Long	*10.24*	*2.28*	*-0.26*
End node	**2039194**	**1732596**	**61150**
Speed (kn)	17.0	17.0	17.0
Average speed AIS (kn)	15.1	12.4	13.8
Aprox speed	15.1	12.4	13.8
SIMROUTE (h)	52.41	22.74	9.14
AIS (h)	58.68	30.32	11.14
SIMROUTE (miles)	886.20	381.81	154.85
AIS (miles)	881.30	378.58	155.28
Savings (miles)	-4.90	-3.23	0.43
Savings (% miles)	-0.56	-0.85	0.28
Savings (h)	6.27	7.58	1.99
Savings (%)	10.68	24.99	17.90

Table 22 – Summary table for Case 4. The total saved time for this round trip is 40 hours approximately, making up a 13.56% of the savings in terms of time. However, there is not such a big difference in distance, with just a 0.22% of savings.

CASE 4	**TOTAL**
SIMROUTE (h)	256.92
AIS (h)	297.23
SIMROUTE (miles)	4321.51
AIS (miles)	4331.20
Savings (miles)	9.69
Savings (% miles)	0.22
Savings (h)	40.30
Savings (% time)	13.56

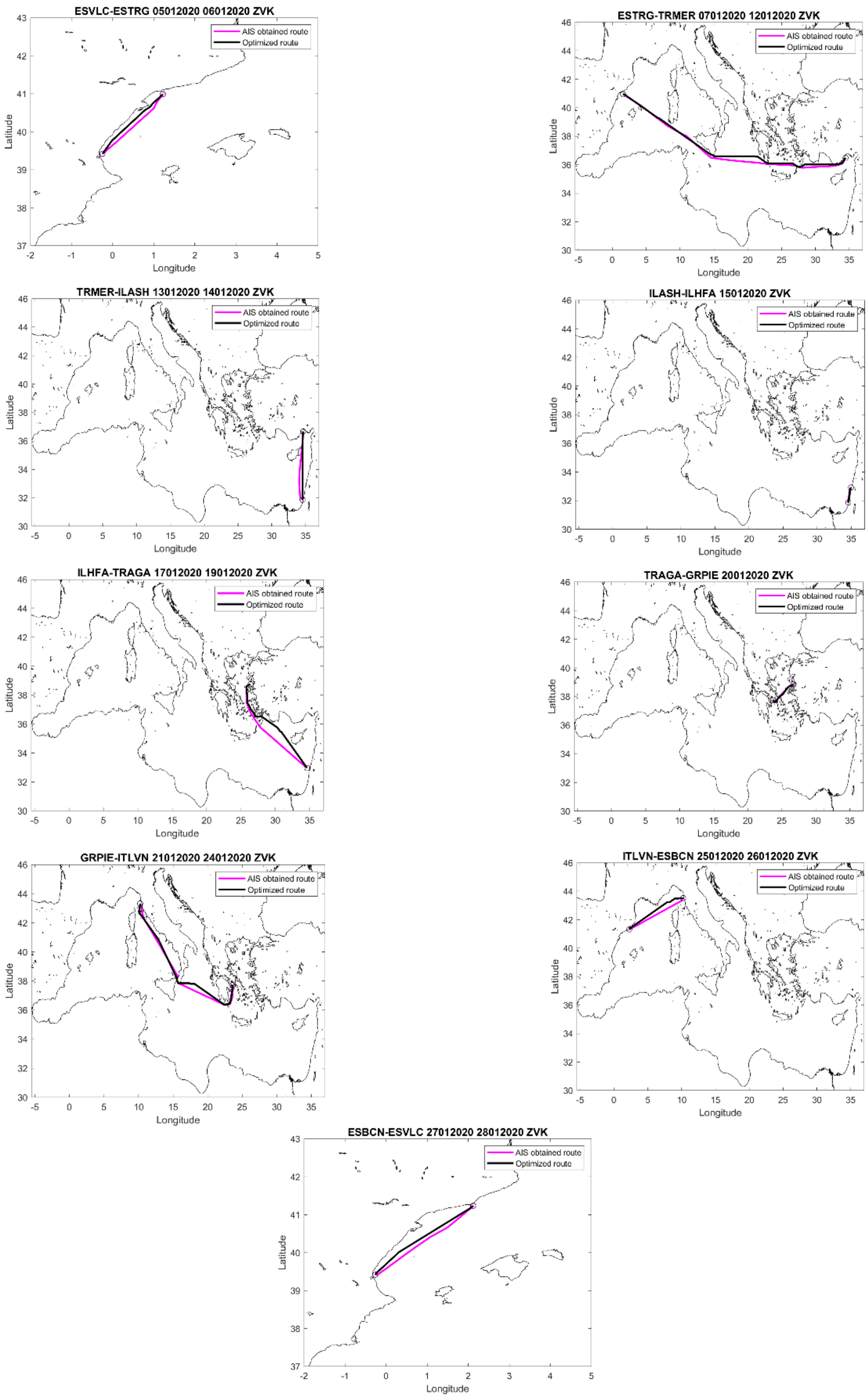

Figure 15 – Case 4 figure compilation of trips. The order is from left to right starting from the upper left corner. The highest saving was in the Ashdod (ILASH) to Haifa (ILHFA) trip with a 39. 41%. On the other hand, the least efficient was the Aliaga (TRAGA) to Piraeus (GRPIE) trip with a -25.71%.

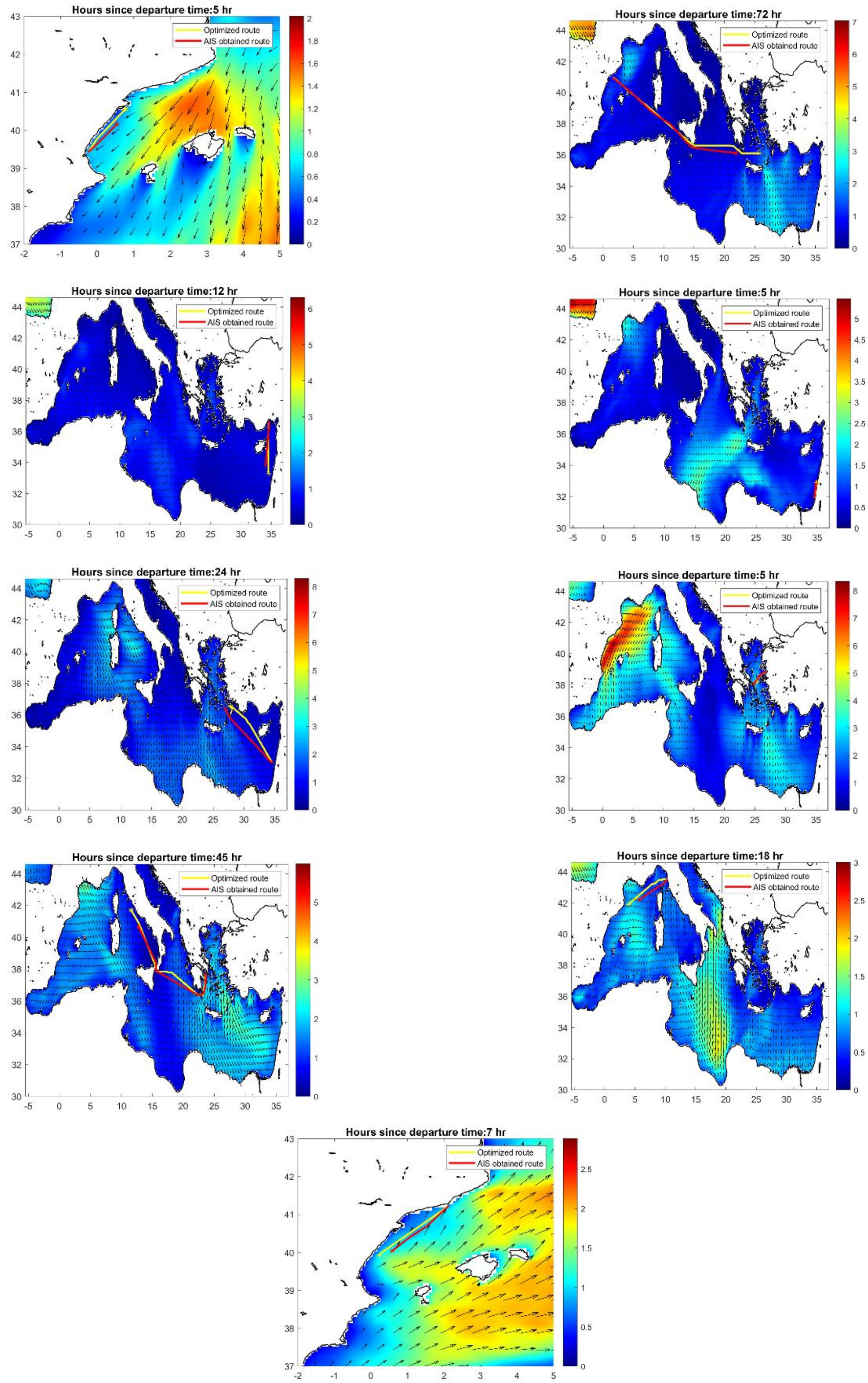

Figure 16 – Case 4 figure compilation of wave maps for the most relevant part of each trip. In case 4 the most relevant wave episodes occur in the first image (ESVLC-ESTRG), and the last one (ESBCN-ESVLC) with also head seas. Source of weather forecasts: COPERNICUS database.

Chapter 4. Discussion

4.1. *find_ports.m* module

As part of the development of the project, the *find_ports.m* module was found to be conflictive when using SIMROUTE. Its simple design did provide with the desired output which were the nodes asked for with the introduced coordinates. However, its integration in SIMROUTE, as a whole, proved to be problematic since the next step in *simroute_opt.m* would take some time to load and would suddenly stop to display the error message 'the introduced node is land'. Therefore, meaning that the user would have to go back to *find_ports.m* and start the process of finding a new node, again.

Although it was not an objective of this work it was thought that the update of this module would improve the 'quality of life' of the user. Now it is possible to obtain both start and end nodes at the same time and check if they belong to 'sea'. It also proved useful the new function of visualization of the wave field map and an AIS route in order to visually check that the obtained nodes correspond to the desired coordinates.

4.2. *AIS_route_analysis.m* module

This module has confirmed the possibility of introducing raw AIS data into MatLab and be able to process it. This means that an available *.csv* file containing information about ship routes obtained from AIS stations or databases can be used for research purposes. This was one of the specific objectives of this work, and the new module serves as a source of information for SIMROUTE users to analyse any desired AIS route.

4.3. *compare_routes.m* module

For an optimum route software such as SIMROUTE it is very interesting to be able to evaluate the similarity of the results with real cases. Being a specific objective of the project, *compare_routes.m* satisfies the need of a module able to process both AIS and optimum routes and compare them, providing the user with valuable information that can be analysed. In addition, since the main objective was to be able to carry out a study of comparison, a case study has been proposed and executed. This tool may prove very useful for future uses of SIMROUTE and, therefore, the following sections are a brief discussion of what the case study has shown, starting in numerical order from Case 1 to Case 4.

Note that Case 1 and 2 consider the average speed of the 'real' ship as SIMROUTE's 'cruise speed' for *simroute_Opt.m*, and Case 3 and 4 consider the ideal speed for the study, which is the optimal 'cruise speed' of 17 kn. The results vary greatly.

4.4. Case 1 29/01/2020 to 18/02/2020

With a maximum saving of 6% and a low at -1.25%, Case 1 does not show major savings when the average speed is used for optimization. This issue is due to the fact that the average speed is not an ideal speed for optimization software, in fact, *simroute_Opt.m* reduces Case 1's speed depending on the incidence and height of the waves to the vessel. Therefore, most of the time the 'virtual ship' is not sailing at an equivalent velocity because the speed shown on AIS data is already been affected by weather.

The raw data obtained from Marine Traffic's service showed that most of the journeys had speeds much lower than the optimal, meaning that the engine load is not the optimal either. It is important to point out then, that varied engine loads will produce results that will be impossible to compare with SIMROUTE 'optimal engine load' ones, because the ability of a real ship to vary their speed overcomes the problem of an optimal route.

A more profound analysis has been done, checking closer the results of every journey:

4.4.1. ESVLC – ESTRG

The difference in miles shows that the saving percentage has been produced by the optimal route with no direct influence of the waves.

4.4.2. ESTRG – TRMER

There have been no savings in terms of time or distance, which means that the optimal route was not good enough. There have not been observed any signs of weather influence on the optimal and AIS route. The most probable cause has been the ability of the real ship to vary their speed, because even though the average speed was 18.75 knots, there have been maximum speeds of up to 20.6 knots.

4.4.3. TRMER – ILASH

In this case, there has not been observed any influence of the waves on the routes. However, the SIMROUTE route has proven to be more efficient with a saving of 20 miles.

4.4.4. ILASH – ILHFA

Once again, there is no influence of the waves, and there appear to be savings in distance and time of 8.60% and 3.65% respectively.

4.4.5. ILHFA – TRAGA

This the first case with an episode of waves, more precisely, there have been head seas of up to 3 – 4.5 m. The AIS route reaches its destination earlier and most probably it is due to the fact that it has been able to vary its speed during the wave episode, with an average speed of 17.2 knots but a progressive increase towards the end of the journey. Moreover, the AIS route has less changes and appears to be more direct than the optimal route produced by SIMROUTE.

4.4.6. TRAGA – GRPIE

With nearly no savings at all, at less than 1% in distance and time, it has not been noticed any influence of the wave field.

4.4.7. GRPIE - ITLVN

This case behaves in the same way as of the previous, the savings are very low and there has not been observed any influence of the wave field.

4.4.7. ITLVN – ESBCN

With no high waves from the wave field, the optimal route aims for a different route while the AIS one takes a more direct one, ending up with no savings. Note that the average speed was 13.7 knots while there were peaks at 14.9 knots.

4.4.7. ESBCN – ESVLC

Finally, with a 1.43% of time saved, it seems there is no direct influence of the wave field due to the low height of the waves. The main difference comes from a closer route to the coast used by SIMROUTE.

4.5. Case 2 30/12/2020 to 25/01/2020

Case 2 has the same issues as Case 1 in terms of varied speed regimes. The maximum saving was 2.90% and the lowest was -48.32%. Again there is special incidence of the change of speed in the real journey because in navigation the vessel is able to vary their speed depending on the weather conditions while the optimization software sets off from a maximum optimal speed that is afterwards reduced by the weather conditions and never increased.

On the one hand, the lowest saving of -48.32% shows an issue of a non-optimal route provided by SIMROUTE for an energetic wave episode that highlights that there need to be some adjustments made on the algorithm.

On the other hand, it is also relevant to say that most of ships already use optimization software or 'optimum routes', especially in fixed round trips, so there may not be great savings obtained from the use of similar optimum routes.

4.5.1. ESVLC – ESTRG

With head seas of less than 1 meter, the AIS route arrives earlier to its destination. The average speed of the real ship was 12.5 knots but had long periods of approximate speeds of 13 knots.

4.5.2. ESTRG – TRMER

In this case, the ship shows no data of the route from Tarragona to Sicily, therefore, the starting point is near the coast of the Italian island. For most of the journey there are beam seas which do not show a direct influence over the routes. Even though the average speed was 13.4 knots, in the last quarter it is steadily increased to up to 15.5 knots.

4.5.3. TRMER – ILASH

This journey has another episode of waves with head seas of up to 3.5 meters. Even though the optimum route is avoiding the biggest part of the bad weather, the AIS route aims for a more direct journey arriving to its destination a bit earlier. There is not a big variation of the real ship's speed during the trip and, therefore, it is very probable that the reduction of the optimum route is caused by the wave episode.

4.5.4. ILASH – ILHFA

The two routes appear to be very similar, no influence from waves is observed on the wave maps. The average speed is 12.9 knots but there is a maximum of 16.9 knots near the end of the route which may reduce the margin for savings of time.

4.5.5. ILHFA – TRAGA

With no direct influence observed from the wave field, SIMROUTE's route was more optimal. The average speed was 19 knots with a maximum of 19.9 knots and a steady decrease to around 18.5 knots towards the end of the trip.

4.5.6. TRAGA – GRPIE

The two routes seem similar with no observable influence of the waves. 16.1 knots of average speed and a decrease of speed of the real route at the end of the trip. There are no remarkable savings.

4.5.7. GRPIE – ITLVN

With an average speed of 17 knots, SIMROUTE’s route was more optimal and avoided wave areas, during a wave episode of waves of 3 meters, head seas. Overall, the optimum route produced a saving of 1.61 %.

4.5.8. ITLVN – ESBCN

This is a very noticeable wave episode of following seas with altitudes of 6.5 – 7 meters. More interestingly, the optimum route sails near the coast of South France to avoid the big waves but the real route reaches its destination 13 hours earlier probably because it profited from the strong following waves and took a more direct route. All this is observed through the savings of this case, -48.32% of losses, showing a direct issue with SIMROUTE’s algorithm in this particular conditions. The effect of this trip on Case 2, as a whole, translates into a big -5.20% of losses in time. Therefore, the conclusion obtained here is not only the need of an improvement of provided data for the analysis but also an improvement of SIMROUTE’s decision for certain situations.

4.5.9. ESBCN – ESVLC

There is no influence of waves (up to 0.5 meters) observed in this trip and there are very similar routes. The average speed is 10 knots.

4.6. Case 3 24/01/2020 to 11/02/2020

Case 3, and Case 4, applies a design speed of 17kn which is the recommended regime of navigation by the ship designer. In this situation there has been a maximum saving of 31.97% and a minimum of -29.80%. This large difference is again due to the actual ship using speeds that are distant from 17kn. For example, in the circumstance of -29.80% and -26.61% the average speed of the ship was 22.2kn and 21.3kn respectively which for long trips is a great difference from 17kn. On the other hand, the opposite occurs for the situations with 31.97%, 28.28%, 19.35% and 12.70% with average speeds of 11.5kn, 12.9kn, 13.9kn and 14.2kn respectively.

Overall, there has been a 2.39% of savings in terms of time and 1.12% in terms of miles.

4.6.1. ESVLC – ESTRG

The negative savings seem to have been produced by the big difference of speed from 17 knots of the optimum route and the 22.2 knots of the real one. However, SIMROUTE's route appears to be more optimal with a saving of 1.3 % in terms of distance.

4.6.2. ESTRG – TRMER

Once again, the negative savings have occurred due to the difference in speeds, being the AIS route average speed 21.1 knots. There has been observed no influence from waves on the routes.

4.6.3. TRMER – ILHFA

In this case, even though the average speed of the AIS route is a bit faster with 17.3 knots, the optimum route reaches its destination earlier and produces savings of 8.83%. There is no noticeable influence from waves.

4.6.4. ILHFA – TRAGA

In the start of this trip there are head seas of 3.5 – 4 meters and it ends with no savings of time due to the real ship having an average speed of 18.3 knots.

4.6.5. TRAGA - GRPIE

With no influence from any wave episode, the savings produced in this case seem to have been produced by the difference of speeds between the optimum (17 knots) and the real (14.2 knots in average).

4.6.6. GRPIE – ITLVN

In this trip, the average speed of the AIS route is 13.9 knots which in contrast with the SIMROUTE's route is slower and less optimal. There is a wave episode of head and beam seas of 4 meters and 5.5 meters respectively before reaching Sicily. Afterwards, there are head seas incident to the optimum route of 4 meters.

4.6.7. ITLVN - ESBCN

In this case, the optimum route is faster and takes a less direct route. The average speed of the real ship is 11.5 knots and there is not any remarkable wave episode.

4.6.8. ESBCN – ESVLC

With no influence from waves, and an average speed of 12.9 knots for AIS route. The optimum route is faster and better and it produces a saving of 28.28 % in terms of time.

4.7. Case 4 05/01/2020 to 28/01/2020

For the last case, the savings range from 39.41% to -25.71%, being -0.13% the only other negative saving. Once again, it can be observed the impact of the differences in speed regimes compared to the cruise speed chosen for the analysis. In the end, to be capable of strictly comparing an analysis to the reality, there are a few conclusions to be discussed.

On the one hand, for the assessing of SIMROUTE as an optimum route software solution, it could be interesting to choose experiments or ships that sail to their optimal engine load, thus, aiming to sail at their design cruise speed, and allowing to see the influence of the wave field on the real ship. Another option could be to access better sources of information that provide the 'aimed' speed for the trip in question so that it can be analysed properly.

On the other hand, SIMROUTE has recently incorporated a script called *make_emissions.m* which calculates the emissions produced by a route. With data about the engine load and regimes it could be relevant to see the benefits of time and emission reduction. Therefore, the design cruise speed can be used to compare this type of situations.

4.7.1. ESVLC – ESTRG

This case, as well as some previous ones, has an optimum route that reaches its destination faster due to the difference in speeds, the average speed of the AIS route is 11.5 knots, and there is no influence of waves during the trip.

4.7.2. ESTRG – TRMER

With an average speed of 14.6 knots, it has a similar situation to the route from Valencia to Tarragona. The optimum route is faster and, thus, it arrives earlier.

4.7.3. TRMER – ILASH

Again, an average speed of 13.2 knots that is lower than the 17 knots which produces a faster 'optimum route' but not really comparable.

4.7.4. ILASH – ILHFA

The AIS route sails with an average speed of 10.9 knots, much lower than the optimum that reaches its destination earlier.

4.7.5. ILHFA – TRAGA

This case shows no savings with a -0.13%, which is probably due to the fact that the average speed is closer to 17 knots and that there are head seas of up to 1.5 – 2 meters.

4.7.6. TRAGA - GRPIE

In this trip, the AIS route is faster with an average speed of 20.4 knots, and no direct influence of wave, it reaches its destination earlier than the optimum route. No savings with a -25.71%.

4.7.7. GRPIE – ITLVN

On the other hand, the AIS route here is slower with an average speed of 15.1 knots. Therefore, the optimum route arrives earlier with no influence from the wave field on either of the ships. The savings have been of a 10.68%.

4.7.8. ITLVN – ESBCN

There is no remarkable influence from waves, the optimum route reaches its destination faster while the average speed of the AIS route is 12.4 knots.

4.7.9. ESBCN – ESVLC

Finally, in this case, with head seas of up to 1.5 meters, SIMROUTE takes a route closer to the coast while the AIS route has an average speed of 13.8 knots, thus, reaching its destination later than the optimum route.

Chapter 5. Conclusions

In this work, the first step has been laid for a future integration of AIS data into optimum software. Three new scripts have been implemented so that it is possible to assess the development of the software and allow to compare its results with real-life cases. In order to make these new additions accessible for other interested students or researchers, there is a short guide on how to use it. In addition, there is a case study for four equal ships that has shown various possible uses for the modules.

To summarize the discussions chapter, there has been an extensive evaluation of 4 cases with the same route and ship characteristics, although Case 1 and 2 have used the average speed of the real routes, and Case 3 and 4 have used the ideal cruise speed of the ships of 17 knots. This means there have been 35 routes analysed in total, for the period from 30/12/2020 to 18/02/2020, and they have shown the different aspects to consider when carrying out a comparison study. First of all, the importance of setting an adequate speed that has not been influenced by any factor in order to obtain real optimum results. Secondly, the fact that a real ship is able to change its velocity or engine load so as to overcome obstacles such as bad weather is a disadvantage for a route optimization software. Thirdly, it is proposed that to evaluate SIMROUTE, the best routes should be those which sail to their optimal speed or cruise speed. Fourthly, that this evaluation can also be done from the emission point of view, observing the savings produced by a route that sails to their optimum speed, even though it may be slower in terms of time but more efficient in terms of pollution.

Table 13 – Results summary for the four different cases analysed.

CASE SUMMARY	CASE 1	CASE 2	CASE 3	CASE 4
SIMROUTE (h)	280.03	272.69	246.05	256.92
AIS (h)	281.08	259.22	252.06	297.23
SIMROUTE (miles)	4362.12	3786.44	4114.25	4321.51
AIS (miles)	4392.01	3746.40	4161.06	4331.20
Savings (h)	1.06	-13.48	6.03	40.30
Savings (% time)	0.38	-5.20	2.39	13.56

This table summary highlights the importance of the objective of each route analysis, however, when evaluating the optimisation against real cases the speed factor is of great influence. A better source of data would provide with information such as the optimal cruise speed and the different optimum speeds for different engine load regimes, allowing to see the raw values of speed with no influence from weather so that it can be evaluated properly with SIMROUTE. Another issue of great importance is found when looking at Case 2 savings value of -5.20%, showing that in some particular conditions such as energetic wave episodes, SIMROUTE still shows calculations that prove to be less optimal in terms of time.

Considering the other uses proposed in this work, the next step will be the development of a guide on how to use the new additions and *make_emissions.m* module to obtain the emissions of a real ship and also assess the results obtained with this tool.

To conclude, as suggestions for future advancements, it would be interesting to see the introduction of a range of speeds or engine loads that *simroute_Opt.m* could choose from when creating an optimal route. This could mean that SIMROUTE would better overcome wave episodes and other disadvantages. Moreover, a better integration of AIS could bring simultaneous optimization for routes by obtaining data on-the-go, for example, with the use of the FNB's AIS station.

References

[1] C. Boren, L. Falevitch, M. Castells-Sanabra, M. Grifoll. 2019. *Added resistance parametrizations due to waves in a weather ship routing.* 1st International Conference of Maritime Science & Technology NASE MORE 2019: Conference proceedings. Dubrovnik, Croatia 17-18th October 2019, pp. 50-59.

[2] C. Padhy. 2008. *Application of wave model for weather routing of ships in the North Indian Ocean.* Nat. Hazards, 44, pp. 373-385.

[3] Copernicus Marine Service database. Available at: https://marine.copernicus.eu/ [Consulted on: 20th of April]

[4] D. Walden. *The Bellman-Ford Algorithm and "distributed Bellman_Ford".* 2003 Available at: https://www.researchgate.net/publication/250014977_THE_BELLMAN-FORD_ALGORITHM_AND_DISTRIBUTED_BELLMAN-FORD [Consulted on: 25th of March]

[5] E. Dijkstra. *A note on two problems in connexion with graphs.* Numerische Mathematik, 1, pp.269-271.

[6] G. Mannarini. 2013. *A Prototype of Ship Routing Decision Support System for an Operational Oceanographic Service.* TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 7 (1), pp. 1-21.

[7] International Maritime Organization. *Automatic Identification Systems (AIS).* Available at: http://www.imo.org/en/OurWork/Safety/Navigation/Pages/AIS.aspx [Consulted on: 30th of March]

[8] International Maritime Organization. *Ship's routeing.* Available at: http://www.imo.org/en/OurWork/Safety/Navigation/Pages/ShipsRouteing.aspx [Consulted on: 30th of March]

[9] J. Del Fante Serra. 2015. *GMDSS: Manual del operador general del GMDSS.*

[10] J. Szłapczyńska, R. and Śmierzchalski. 2009. *Multicriteria Optimization in Weather Routing.* TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 3 (4), pp. 393-400.

[11] K. Takashima, B. Mezaoui, and S. Ruri. 2009. *On the Fuel Saving Operation for Coastal Merchant Ships using Weather Routing.* TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 3 (4), pp. 401-406.

[12] L. Walther, A. Rizvanolli, M. Wendebourg, C. Jahn. *Modeling and Optimization Algorithms in Ship Weather Routing.* International Journal of e-Navigation and Maritime Economy. 2016, Abstract. Available at: https://www.sciencedirect.com/science/article/pii/S2405535216300043 [Consulted on: 25th of March]

[13] Ll. Basiana. 2017. *Study of the feasibility of the ship-weather routing algorithm SIMROUTEv2 on short sea shipping routes.* University of Strathclyde undergraduate thesis.

[14] M. Grifoll, and F.X. Martínez de Osés. 2016. *A Ship Routing System Applied at Short Sea Distances.* Journal of Maritime Research, Vol. XIII. No. II (2016), pp 3-6.

[15] Marine Traffic Database. Available at: https://www.marinetraffic.com/ [Consulted on: 6th of April]

[16] M.H. Simonsen. 2015. *State-of-art within ship routing.* ASME 24th International Conference on Ocean, Offshore and Arctic Engineering. Available at: https://asmedigitalcollection.asme.org/OMAE/proceedings-abstract/OMAE2015/56499/V003T02A053/280363 [Consulted on: 25th of March]

[17] Puertos del Estado Wave Map database. Available at: http://opendap.puertos.es/thredds/catalog.html [Consulted on: 6th of April]

[18] R. Dechter, and J. Pearl. 1985. *Generalized best-first search strategies and the optimality of A*.* Journal of the ACM, 32 (3), pp. 505-536.

[19] S. Wei, and P. Zhou. 2012. *Development of a 3D Dynamic Programming Method for Weather Routing.* TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 6 (19, pp. 79-85.

[20] UPC. *SIMROUTE TECHNICAL MANUAL. SIMROUTE: a weather ship routing software for academic purposes.*

[21] UPC. *SIMROUTE USER'S MANUAL. SIMROUTE: a weather ship routing software for academic purposes.*

[22] Vessel Performance Optimization Magazine. *The importance of weather routing in fuel-efficient shipping.* Available at: https://vpoglobal.com/2018/09/14/the-importance-of-weather-routing-in-fuel-efficient-shipping/ [Consulted on: 25th of March]

[23] X. Calvo, M. Grifoll, F.X. Martínez de Osés, M. Castells. 2018. *Automatic Identification system data implementation to a weather ship routing.* 19th Annual General Assembly of the International Association of Maritime Universities: IAMU Students Conference Proceedings. Barcelona, Spain. 17-19th October 2018, pp. 21-26.

Annex 1. AIS_route_analysis.m

```
close all;
clear all;
clc;

%%% LOADING OF FILES

%[t(yyyy-MM-dd HH:mm:SS) Source Speed Course LAT LONG] (CSV)

formatspec='%{yyyy-MM-dd HH:mm:SS}D%f%f%f%f%f';
filename='2901.csv' %AIS route

%%% END OF LOADING %%%%%%%%%%%%%%%%%%%%%%%%

FILE=readtable(filename);

departure=FILE.x_Timestamp(1)
arrival=FILE.x_Timestamp(end)

t=datenum(FILE.x_Timestamp(end)-FILE.x_Timestamp(1)).*24; %Calculate time of
the trip in hours
t=fix(t);
FILE.x_Timestamp=datenum((FILE.x_Timestamp(:)-FILE.x_Timestamp(1)).*24);

route=[FILE.x_Timestamp,FILE.Speed,FILE.Course,FILE.Latitude,FILE.Longitude];

xx=(0:0.083:t)';
finish=route(end,:);

route=[xx,interp1(FILE.x_Timestamp,FILE.Speed,xx,'method','extrap'),interp1(F
ILE.x_Timestamp,FILE.Course,xx,'method','extrap'),interp1(FILE.x_Timestamp,FI
LE.Latitude,xx,'method','extrap'),interp1(FILE.x_Timestamp,FILE.Longitude,xx,
'method','extrap');finish];
LatRad=deg2rad(FILE.Latitude);
LongRad=deg2rad(FILE.Longitude);

h=numel(LatRad)-1;
        for k=1:1:h
        A(k,1)=sin(LatRad(k)).*sin(LatRad(k+1));
        B(k,1)=cos(LatRad(k)).*cos(LatRad(k+1)).*cos(LongRad(k+1)-
LongRad(k));
        end

Cosd=A+B;
d=rad2deg(acos(Cosd))*60;
dTOTAL=sum(d);
averageSpeed=mean(FILE.Speed);

disp(['Total time in reality, obtained from AIS data   ' num2str(route(end,1))
'   hours'])

disp(['Distance obtained from AIS data                 ' num2str(dTOTAL) '
miles'])
```

```
disp(['Average speed of                                      ' num2str(averageSpeed)
'  kn'])

disp(['Port of origin                                        ' num2str(route(1,4))
'° ' num2str(route(1,5)) '°'])

disp(['Port of destination                                   ' num2str(route(end,4))
'° ' num2str(route(end,5)) '°'])

return
```

Annex 2. Find_ports.m

```
% Tool to find final/end nodes from the mesh accordingly lon/lat from a
% Port. Allows to see the limits of the wave map and be
% able to choose a correct node for your port.

close all;
clc;
clear all;

%Introduce mesh information file
load('in/nods.mat');
%Introduce wave field file
load in/waves_20200127_2801CP %wave information

%%%%Load wave conditions (with or without modifications).
%Would you like to display the wave field? Y=1 N=0
WF=1;

%Would you like to display the AIS route? Y=1 N=0
AISR=0;

%Load the AIS route
if AISR==1;
    filename='0501_0601.csv'
end

%Introduce coordinates of a Port:

%Valencia
Lon_port1=-0.24;
Lat_port1=39.43;

%Tarragona
Lon_port2=1.19;
Lat_port2=41.0164;

%%% End of User Inputs


if WF==1;
    [LON,LAT]=meshgrid(tira_lon,tira_lat);
    %nt=size(hs);nt=nt(2);
    %nt=ceil(qfix.cost_fix(end));
    Hmax=max(max(max(hs)));

    Hs=squeeze(hs(:,1));%
    Dir=squeeze(dir(:,1));
    Hs_r=reshape(Hs,Nx,Ny);
    Dir_r=reshape(Dir,Nx,Ny);
    pcolorjw(LON,LAT,Hs_r');
    caxis([0 Hmax])
    colorbar,shading flat,hold on,colormap('jet')
    Hy=squeeze(Hs_r.*cosd(Dir_r));
    Hx=squeeze(Hs_r.*sind(Dir_r));
end
```

```
%Node of origin
Node_port1=search_nods(Lon_port1,Lat_port1,4);
hold on

load in/ldc_euro_i_mask;
plot(lon,lat,'r-','linewidth',1)
h1=plot(nodes(Node_port1,1),nodes(Node_port1,2),'+k','MarkerSize',20);
xlim([LonMin LonMax]);
ylim([LatMin LatMax]);

%Node of destination
Node_port2=search_nods(Lon_port2,Lat_port2,4);
h2=plot(nodes(Node_port2,1),nodes(Node_port2,2),'+g','MarkerSize',20);

xlim([LonMin LonMax]);
ylim([LatMin LatMax]);

if AISR==1;
    FILE=readtable(filename);

    t=datenum(FILE.x_Timestamp(end)-FILE.x_Timestamp(1)).*24; %Calculate time
of the trip in hours
    t=fix(t);
    FILE.x_Timestamp=datenum((FILE.x_Timestamp(:)-FILE.x_Timestamp(1)).*24);


route=[FILE.x_Timestamp,FILE.Speed,FILE.Course,FILE.Latitude,FILE.Longitude];

    plot(route(1,5),route(1,4),'om')
    hold on
    plot(route(end,5),route(end,4),'om')
    plot(route(:,5),route(:,4),'-m','linewidth',2)
end

legend([h1 h2],'Origin','Destination');

if isnan(hs(Node_port1,1))
    disp('The node of ORIGIN is land')
    return
end
if isnan(hs(Node_port2,1))
    disp('The node of DESTINATION is land')
    return
end

disp(['The port of ORIGIN      coordinates are  ' num2str(Lat_port1) 'º  '
num2str(Lon_port1) 'º']);
disp(['                               node is   ' num2str(Node_port1)])

disp(['The port of DESTINATION coordinates are  ' num2str(Lat_port2) 'º  '
num2str(Lon_port2) 'º']);
disp(['                               node is   ' num2str(Node_port2)])
```

Annex 3. Compare_routes.m

```
close all;
clear all;
clc;

%%% LOADING OF FILES

%ROUTES MUST BE INTRODUCED IN THE FOLLOWING FORMAT:
%[t(yyyy-MM-dd HH:mm:SS) Speed Course Heading LAT LONG] (CSV)
%[t(yyyy-MM-dd HH:mm:SS) Source Speed Course LAT LONG] (CSV)

%                                           (INPUT)

nom='Test';
formatspec='%{yyyy-MM-dd HH:mm:SS}D%f%f%f%f';
filename='0501_0601.csv' %AIS route (Preferably files of the type .csv)
q=load('0501_0601simroute.mat'); %Optimized SIMROUTE route

%LonMin=-5.5;LonMax=37.0;LatMin=30;LatMax=46; %Mediterranean
%LonMin=-5.5;LonMax=18.0;LatMin=37;LatMax=44.6; %Spain to Italy
LonMin=-2;LonMax=5.0;LatMin=37;LatMax=43;%Catalan countries

%%EUROPE MASK
load in/ldc_euro_h_mask;
%load in/ldc_euro_mask;
%load in/ldc_euro_i_mask;

%%% END OF LOADING %%%%%%%%%%%%%%%%%%%%%%%%

[fPath, fName, fExt] = fileparts(filename);

%Start of operations


FILE=readtable(filename);

t=datenum(FILE.x_Timestamp(end)-FILE.x_Timestamp(1)).*24; %Calculate time of
the trip in hours
t=fix(t);
FILE.x_Timestamp=datenum((FILE.x_Timestamp(:)-FILE.x_Timestamp(1)).*24);

route=[FILE.x_Timestamp,FILE.Speed,FILE.Course,FILE.Latitude,FILE.Longitude];

xx=(0:0.083:t)';
finish=route(end,:);

route=[xx,interp1(FILE.x_Timestamp,FILE.Speed,xx,'method','extrap'),interp1(F
ILE.x_Timestamp,FILE.Course,xx,'method','extrap'),interp1(FILE.x_Timestamp,FI
LE.Latitude,xx,'method','extrap'),interp1(FILE.x_Timestamp,FILE.Longitude,xx,
'method','extrap');finish];

LatRad=deg2rad(FILE.Latitude);
LongRad=deg2rad(FILE.Longitude);
```

```
h=numel(LatRad)-1;
for k=1:1:h
    A(k,1)=sin(LatRad(k)).*sin(LatRad(k+1));
    B(k,1)=cos(LatRad(k)).*cos(LatRad(k+1)).*cos(LongRad(k+1)-LongRad(k));
end

Cosd=A+B;
d=rad2deg(acos(Cosd))*60;
dTOTAL=sum(d);
averageSpeed=mean(FILE.Speed);
savingshour=route(end,1)-q.cost_opt;
savingsdistance=dTOTAL-q.length_opt;
savingspercentagedist=savingsdistance*100/dTOTAL;
savingspercentage=savingshour*100/route(end,1);

plot(route(1,5),route(1,4),'om')
hold on
plot(route(end,5),route(end,4),'om')
h1=plot(route(:,5),route(:,4),'-m','linewidth',2);

xlabel('Longitude'),ylabel('Latitude')
plot(lon,lat,'k-')
nred=2;
axis([LonMin LonMax LatMin LatMax]);

set(gcf,'PaperPositionMode','Auto')

pos=find(q.costt<t);pos_ok=pos(end);

plot(q.nodes(q.nods_trip(end),1),q.nodes(q.nods_trip(end),2),'ok')
plot(q.nodes(q.nods_trip(1),1),q.nodes(q.nods_trip(1),2),'ok')
h2=plot(q.nodes(q.nods_trip(1:end),1),q.nodes(q.nods_trip(1:end),2),'-
k','linewidth',2);

legend([h1 h2],'AIS obtained route','Optimized route')
tit=nom,box on
title(tit,'fontsize',12,'fontweight','bold')

disp(['Total time with optimization                     ' num2str(q.costt(end))
'    hours'])

disp(['Distance with optimization                       ' num2str(q.length_opt)
' miles'])

disp(['Total time in reality, obtained from AIS data    ' num2str(route(end,1))
'    hours'])

disp(['Distance obtained from AIS data                  ' num2str(dTOTAL) '
miles'])

disp(['Average speed of                                 ' num2str(averageSpeed)
'   kn'])

disp(['Saved                                            ' num2str(savingshour)
'   hours'])

disp(['Saved                                            '
num2str(savingspercentage) '    % (h)'])
```

```
disp(['Saved                                                    '
num2str(savingsdistance) '   miles'])

disp(['Saved                                                    '
num2str(savingspercentagedist) '   % (miles)'])

disp(['Port of origin                                           ' num2str(route(1,4))
'° ' num2str(route(1,5)) '°'])

disp(['Port of destination                                      ' num2str(route(end,4))
'° ' num2str(route(end,5)) '°'])

set(gcf,'PaperPositionMode','Auto')
nom2=['out/compare_routes figures/' nom];
print ('-dpng','-r300',nom2)
```

Annex 4. SIMROUTE User’s and Technical Manuals

SIMROUTE USER'S MANUAL

SIMROUTE: A WEATHER SHIP ROUTING SOFTWARE FOR ACADEMIC PURPOSES

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Supported by THE NIPPON FOUNDATION

Table of Contents

1 WHAT IS SIMROUTE? 5

2 SIMROUTE AS A TEACHING TOOL (STCW FRAMEWORK) 5

3 SIMROUTE SCRIPTS 6

A. START.M 7
B. MAKE_MESH.M 7
C. MAKE_WAVES.M 7
D. SIMROUTE_OPT.M 7
E. SIMROUTE_DMIN.M 8
F. MAKE_MESH_MED.M 9
G. MAKE_WAVES_MED.M 9
H. MAKE_SEA_STEP1.M 9
I. MAKE_SEA_STEP2.M 9
J. PLOT_ROUTES.M 10
K. PLOT_ROUTES_AND_WAVES.M 10
L. PLOT_WAVES_FROM_NC.M 10
M. PLOT_WAVES_INTERPOLATED.M 10
N. MAKE_EMISSIONS 10

4 SIMROUTE FLOW CHART 12

5 USING THE PROGRAM STEP BY STEP 13

6 CASE TEST 15

List of figures

Figure 1: Simroute flow chart 12

Figure 2: start script 15

Figure 3: make_mesh script 16

Figure 4: Mesh made by selecting maximum and minimum longitudes and latitudes . 17

Figure 5: make_waves.m script 17

Figure 6: Simroute_Opt.m script 19

Figure 7: find_ports script to find start node 20

Figure 8: find_ports script to find end node 21

Figure 9: simroute_Opt.m script with start and end nodes introduced 22

Figure 10: variables shown in Variables window 23

Figure 11: simroute_Dmin script 23

Figure 12: plot_routes.m script 24

Figure 13: optimum and minimum distance routes plotted 25

Figure 14: plot_routes_from_nc.m 26

Figure 15: plot_routes_from_nc.m (waves height and direction) 26

Figure 16: plot_routes_interpolated.m 27

Figure 17: plot_routes_interpolated.m 27

Figure 18: plot_routes_and_waves.m (minimum distance route in black, optimized route in magenta) 28

Figure 19: make_Emissions.m 29

Figure 20: make_Emissions.m results 31

By: The SIMROUTE team

Barcelona School of Nautical Studies (FNB)
Universitat Politècnica de Catalunya – BarcelonaTech
Pla de Palau, 18, 08003 Barcelona, Spain
e-mail: simroute@fnb.upc.edu

SIMROUTE INITIATION MANUAL

1 WHAT IS SIMROUTE?

SIMROUTE is software designed for maritime route optimizations which carries out simulations of short and long distances. This software has been developed by a team of researchers at the Universitat Politècnica de Catalunya (UPC-BarcelonaTech) in Matlab language. The spatial scope of SIMROUTE is the Mediterranean Sea, fully or partially selected, depending on the longitudes and latitudes introduced by the user. In order to run the program, there are several parameters to introduce: port of departure, port of arrival, vessel speed and the wave field for the days of the voyage to simulate.

SIMROUTE is a program composed by several scripts and Matlab functions which, when executed sequentially, allow for the definition of a computational mesh according to the selected longitudes and latitudes. Furthermore, it also allows for the definition of start and end nodes in order to carry out the route optimization between these two nodes.

The SIMROUTE's main and most used components are described herein together with the required inputs for each of them.

2 SIMROUTE AS A TEACHING TOOL (STCW FRAMEWORK)

The pedagogic purpose of this software is to provide skills on ship routing optimization, to assess the impact of the meteo-oceanographic variables (such as wind, waves or currents) on ship navigation and to highlight the relevance of ship routing in terms of sailing time, fuel consumption and harmful emissions for the environment. This software is very novel in the framework of teaching innovation in MET institutions.

From an academic point of view, this software deals some specific topics that are part of the syllabi of Maritime Education Training (MET) institutions' programs and that are included in STCW 95/2010 Code (IMO 2011). The STCW Convention 1978 has been amended by the 2010 Manila Amendments and articulates the minimum standards of competence required for seagoing personnel in detail in a series of tables (part A of the Code).

The academic modules that Simroute software includes meet the requirements of the STCW Code and provide schools with a modern way to assess STCW competences. A particular emphasis is done in terms of air emissions and safety of navigation. For instance, this software could be included in existing study programs of the Bachelor's and Master's degrees of Nautical Science and Maritime Transport in subjects such as "Routes and Compasses," "Nautical Meteorology and Oceanographic," or "Bridge Procedures" among others. The software could be also used to assess some specific topics related to Marine Environment Protection.

In a more general context, this software should be considered as a progressive step forward to develop best practices for a more efficient and sustainable maritime industry and to provide maritime industry with better-educated seafarers.

3 SIMROUTE SCRIPTS

The scripts used by SIMROUTE are the following ones, which are listed in the correct order of launching:

a. start.m
b. make_mesh.m
c. make_waves.m
d. simroute_Opt.m
e. simroute_Dmin.m

a. start.m

This is an initializing script that links the required folders which will be used afterwards.

b. make_mesh.m

This script creates the Mediterranean Sea computational mesh defining the maximum and minimum longitudes and latitudes. Furthermore, it can determine the computational mesh resolution results by altering the quantity of nodes as desired.

c. make_waves.m

This script uses the *opendap* data and the file created by *make_mesh.m* for creating a .mat file with the waves and their direction which are interpolated at the created mesh. Furthermore, it defines the vessel's time of departure from the selected port of origin. Therefore, it is necessary to define the initial time of the trip and the name of the file (or files) downloaded from the *opendap*.

When looking at the *make_waves.m* file, the use of the correct wave files has to be ensured. The wave files (in *netcdf* format), downloaded from *Puertos del Estado* website, (see direction in the .m file header), have to be introduced in the *make_waves.m* script. Each file poses the waves conditions (wave height and direction) of one entire day (hourly data). For instance, if the path of the route takes 80 hours and only 3 wave scripts (accounting for 72 hours) have been introduced, the software will reply "error". Once the correct wave files are introduced and the estimated time of departure (ETD) is established, an output wave script is obtained and it is used for running, *simroute_Opt.m* file.

d. Simroute_Opt.m

This main script calculates the optimum route taking into account the wave information interpolated from start and end nodes. When *simroute_Opt.m* is executed the initial speed of the vessel (v_0) is modified taking into account the wave field of the optimized function cost (i.e. sailing time). Previously, using *find_ports.m* script the initial and end nodes (ports) may be easily determined. When the user introduces the latitude and longitude of a point on the map, this function gives the node these coordinates represented on the mesh.

The required inputs for *simroute_Opt.m* are:

- Start node
- End node
- Name of the file generated by *make_waves.m*
- Vessels cruising speed
- Define whether Dijkstra o A* algorithm is going to be used (algorithms description included in SIMROUTE Technical Manual).

The function *simroute_Opt.m* lets the user know whether the selected node is situated at land or at sea.

e. **simroute_Dmin.m**

Taking advantage of the files generated by *simroute_Opt.m* and *make_waves.m*, this script carries out the calculations for the minimum distance route to be compared with the optimized routes.

Through the output scripts that *simroute_Opt.m* creates, *simroute_Dmin.m* can be run and the results of the shortest path route with and without waves, in hours, will be obtained.

In addition to the aforementioned scripts, there are 4 scripts which are very useful when carrying out calculations for the long-distance scenarios where a mesh including all Mediterranean Sea is used. These scripts are described hereby:

f. make_mesh_MED
g. make_waves_MED
h. make_sea_step1
i. make_sea_step2

f. make_mesh_MED.m

This script works in the same way as *make_mesh.m* but the maximum and minimum latitudes and longitudes which limit the Mediterranean Sea have already been introduced. Resolution can still be defined as in *make_mesh.m*.

g. make_waves_MED.m

This script works as *make_waves.m* except that the time of departure is fixed at 00:00h and cannot be changed.

h. make_sea_step1.m

Together with *make_sea_step2.m*, this script is used to open the straits zones where there should be water but there is not because of the interpolation made by *make_waves.m*.

make_sea_setp1.m allows the user to select which part of the Mediterranean Sea is going to be analyzed.

i. make_sea_step2.m

After executing former script, the *make_sea_step2.m* allows the user to define the specific rectangle inside zone designated on the first step where the new and more detailed interpolation is going to be done in order to see the wave fields on those points.

The scripts *make_sea_step2_ew.m* and *make_sea_step2_ns.m* can be differentiated by means of the direction of interpolation on the rectangle, being the first one done from east to west and the second one from north to south (that is opening a channel horizontally or vertically).

Furthermore, there are 4 very useful functions for viewing the results:

j. plot_routes
k. plot_routes_and_waves
l. plot_waves_from_nc
m. plot_waves_interpolated

j. **plot_routes.m**

When the user gives to this script the resulting files from applying *simroute_Opt.m* and *simroute_Dmin.m*, this script carries out the representation of both routes on the zone of the Mediterranean Sea previously defined on *make_mesh.m*. It only represents the coast line and the routes.

k. **plot_routes_and_waves.m**

This second function represents and saves the wave fields evolution together with the route in an hourly manner until getting to the end node.

l. **plot_waves_from_nc.m**

This function plots the wave field when a *netcdf* file is given.

m. **plot_waves_interpolated.m**

This function plots the interpolated wave fields.

There is another function related to the emission of pollutants, described below:

n. **Make_Emissions**

This function has been developed to calculate the fuel consumption and the emission of pollutants per trip (SO_2, NO_x, CO_2 and PM).

When the user gives to this script the resulting files from applying *simroute_Opt.m* and *simroute_Dmin.m*, this script carries out the calculation of above parameters.

The user has to introduce a list of variables which are ship specific, as explained in paragraph 1.4 of the Simroute Technical Manual.

4 SIMROUTE FLOW CHART

What has been stated in former paragraphs can be summed up in a flow chart as the one shown below. This flow chart shows the order of execution, together with the names of the scripts and functions which control different actions.

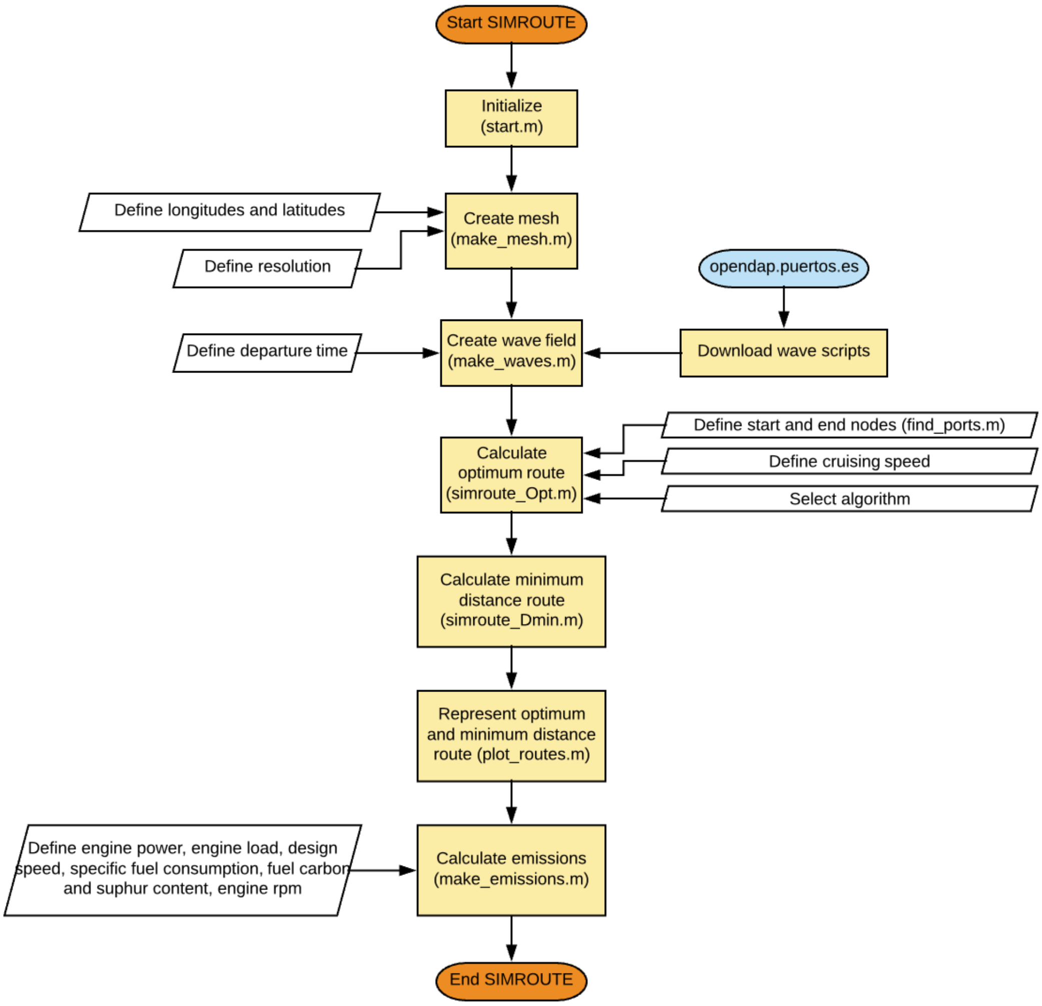

Figure 1: Simroute flow chart

5 USING THE PROGRAM STEP BY STEP

1. Download documents with .m extension copying Dropbox folders (ie., Folders: IN, OUT, TOOLS_PLOTS, FUNMATLAB) together with the rest of .m files.

   - IN folder: composed by 5 general program input files. In this folder, there is also the .nc files (waves files) which were downloaded from the Puertos del Estado webpage: http://opendap.puertos.es/thredds/catalog.html
   - OUT folder: 4 files which are generated as program outputs.
   - TOOLS_PLOTS folder: 4 files for graphic tools.
   - FUNMATLAB folder: 12 files of functions used by the program.

2. From the Puertos del Estado web page, search the wave file to be analyzed that will be used in the simulation.
   The wave data available in this web page goes from June 2016 to October 2018.
   For example:
   - Go to folder: oleaje/Gran escala/Mediterraneo
   - Select year and month.
     The files to download are the ones with short names. This name indicates year, month and day with the extension .nc; for instance, HW-20170125-HC.nc indicates that the file will show the wave height for the 25th of January 2017.
   - When in the file, download the HTTPServer and save the file (or files) into the IN folder. If the route that is going to be simulated is long and cannot fit in on file, more files will have to be downloaded.
     This can be visualized by going to Viewers: Godiva2 (browser-based) (note: open with Chrome in case Explorer doesn't work). At top left menu select HW-20170125-HC.nc to select the variable to visualize. Click on *auto* inside the graphic in order to view the animation.

3. Open MATLAB. *OPEN start.m* file. This file allows the access to different folders. Once *RUN* is activated, which runs the program, the question *Change Folder* appears. The folder needs to be changed because all the files that Matlab can read will appear on left menu.

4. Open *make_mesh.m* file. If *RUN* is activated, the program will draw the mesh for further calculations. In this situation, the latitude and longitude points can be changed in order to get different mesh types.

In the file *make_waves.m*, there is a variable named Tini_trip; this variable is used to specify the vessel's time of departure related to the first day interpolated in the wave files. When Tini_trip=0, the vessel's departure will be at 00:00; if Tini_trip=3, the vessel will leave at 03:00.

5. Open and run *make_waves.m* in order to open wave files previously saved. At line 9: ARX={'HW-20161121-HC.nc','HW-20161122-HC.nc'}, the user can change the files by introducing the name of the files the user is interested in analyzing.

6. Open and run *simroute_Opt.m* in order to look for the optimum route taking into account the mesh and the waves that have been introduced. Be aware that the name of the wave file which is being analyzed has to be loaded at line 10. Furthermore, if the route is changed, the start and end nodes of the route have to be changed too. In order to do so, the user has to open the file *find_ports.m* in the folder *Tools_plot*. Here is where the port coordinates have to be introduced.

7. Open and run *simroute_Dmin.m*. In this case, the shortest path is calculated without the effect of waves.

8. Open and run *make_Emissions.m*. In this script, there are several ship specific variables which have to be introduced by the user. These variables are the installed engine power, the engine load, the design speed of the vessel, the specific fuel consumption, the engine speed of rotation and the fuel characteristics. Once the variables have been introduced, the script is run and it outputs the fuel consumption and the quantity of pollutants emitted into the atmosphere, both for the minimum distance and for the optimized route.

It is recommended to save a folder with all the original files and copy the folder in order to make tests.

6 CASE TEST

In order to make what was previously stated more understandable, below is a detailed example of the way to proceed to undertake a simulation.

The area of study is the Western Mediterranean and the type of vessel used is a Ro-Ro vessel sailing from Barcelona to Genève.

In order to run the simulation, the first thing the user determine sure is that he/she has downloaded the wave scripts of the dates he/she is willing to analyze (see Part 4 point 2 of present manual). Afterwards, proceed as follows:

1) Open Matlab and call *start.m* (enter) in Command Window

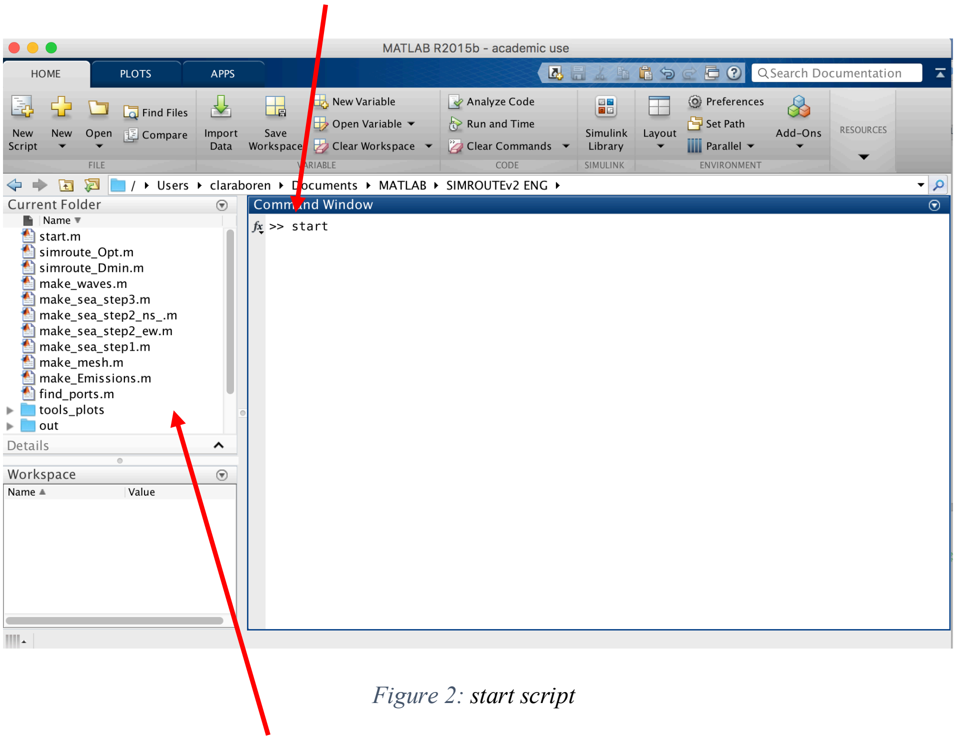

Figure 2: start script

By doing so, the user will have access to the different folders (showed in grey, “non-active”, before and in black, “active”, after).

2) Call *make_mesh* script in Editor mode by double clicking on it in Current Folder window in order to introduce the required inputs.

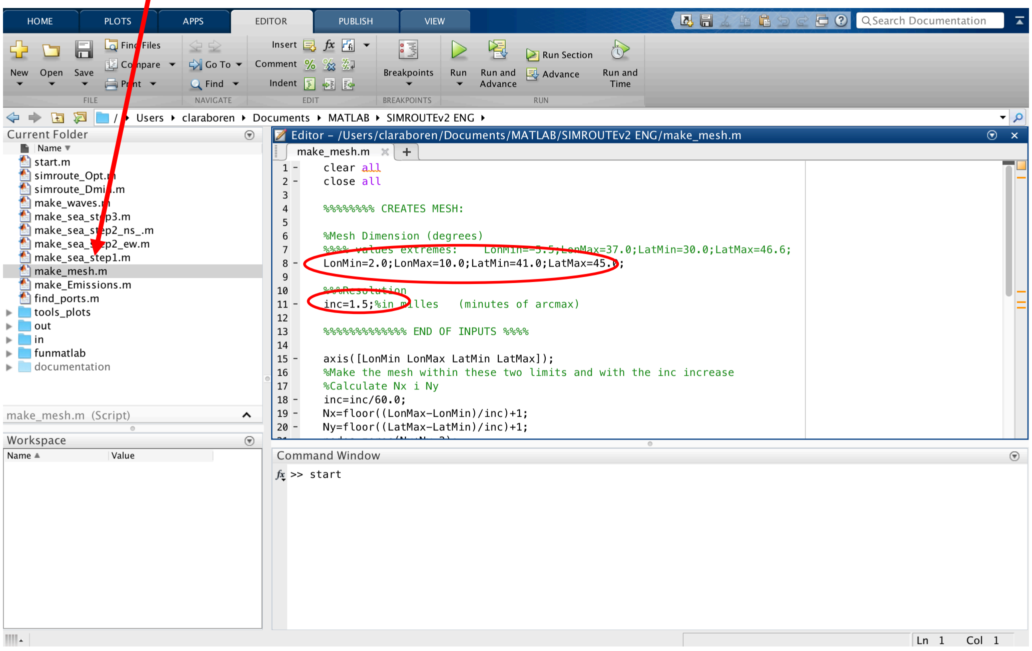

Figure 3: make_mesh script

On the Editor window, the user has to introduce the minimum and maximum longitudes and latitudes of the mesh he/she wants to create. Obviously, this mesh has to enclose latitudes and longitudes of port of departure and port of arrival.

In our example the selected longitudes and latitudes are the following (as seen in figure 3 above):

LonMin=2.0 LonMax=10.0 LatMin=41.0 LatMax=45.0

The user has to select the mesh interval too (in miles). This means the distance between the computational nodes. As seen in figure 3, the mesh interval in our example is 1.5 miles (line 11 of the script, inc=1.5).

At this point, RUN the script and the mesh will be made.

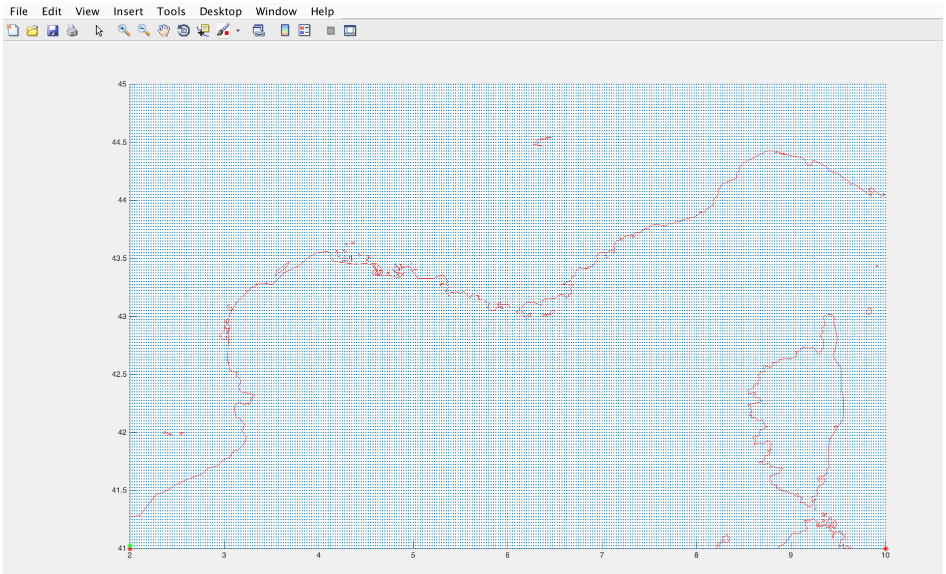

Figure 4: Mesh made by selecting maximum and minimum longitudes and latitudes

3) Call *make_waves* script in Editor mode by double clicking on it in the Current Folder window.

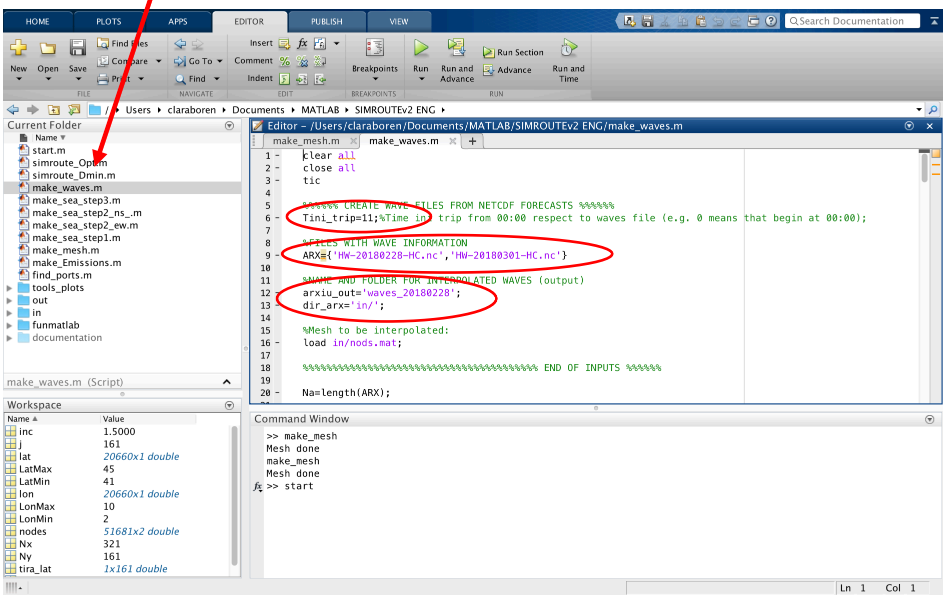

Figure 5: make_waves.m script

In order to interpolate the waves field into the computational mesh, previously the user has to have downloaded the wave files from Puertos del Estado web page (as shown in part 4 of User's Manual).

In the example shown in these pages, the files analyzed where the ones for the 28th of February and the 1st of March 2018.

The user has to introduce the starting time of the trip represented by Tini_trip variable in this program (line 6 of *make_waves* script, figure 5). This means selecting the start time of the wave field. The user has to bear in mind that giving the value 0 to this variable would mean midnight.

Afterwards, the user has to introduce the name of the wave fields to be interpolated, which would be the following ones in our example (line 9 *make_waves* script, figure 5):

ARX={'HW-20180228-HC.nc','HW-20180301-HC.nc'}

The user has to give a name to the interpolated output file. In our example, the output file is called 'waves_20180228', as stated in line 12 of *make_waves* script of figure 5 (arxiu_out='waves_20180228').

Now the user has to RUN 

the script and the interpolated wave field will be generated.

The generated file will be found in the folder IN as per below order:

dir_arx='in/'   (line 13 figure 5)

4) Call *simroute_Opt.m* script in Editor mode by double clicking on it in the Current Folder window.

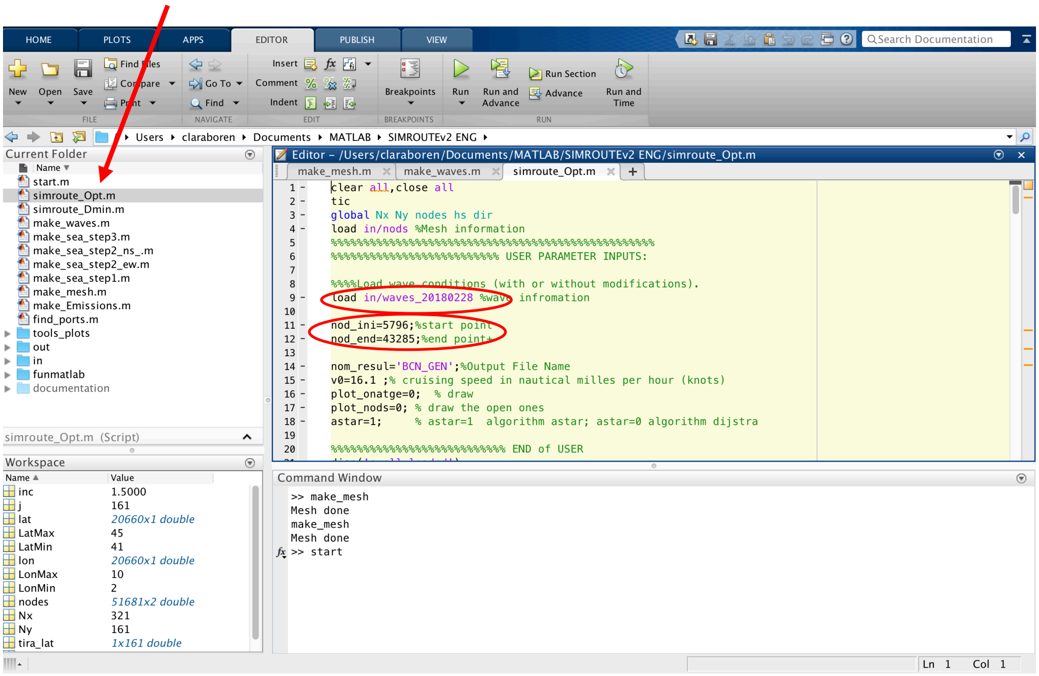

Figure 6: Simroute_Opt.m script

By running *simroute_Opt.m* script, the optimum route can be found taking into account the effect of waves. Nevertheless, before running it, the user has to ask *simroute_Opt.m* script to load the waves interpolated file which is located in IN folder.

Load in/waves_20180228 (see line 9 of figure 6).

In this script, the cruising speed of the vessel can be changed (variable v0, line 15 figure 6).

In the same script, the user has to indicate the start (nod_ini) and end (nod_end) nodes (see line 11 and 12 of figure 6) and he/she also has to give a name to the output file, as shown below:

nom_resul='BCN-GEN'; (line 14 figure 6)

In order to indicate the start and end nodes, the user has to find out which of the nodes of the mesh he/she has created are situated in the closest position to the Port of Departure and Port of Arrival of the simulated route. These positions can be found out by using the *find_port.m* script.

5) Call *find_ports.m* in Editor mode by double clicking on it in the Current Folder window.

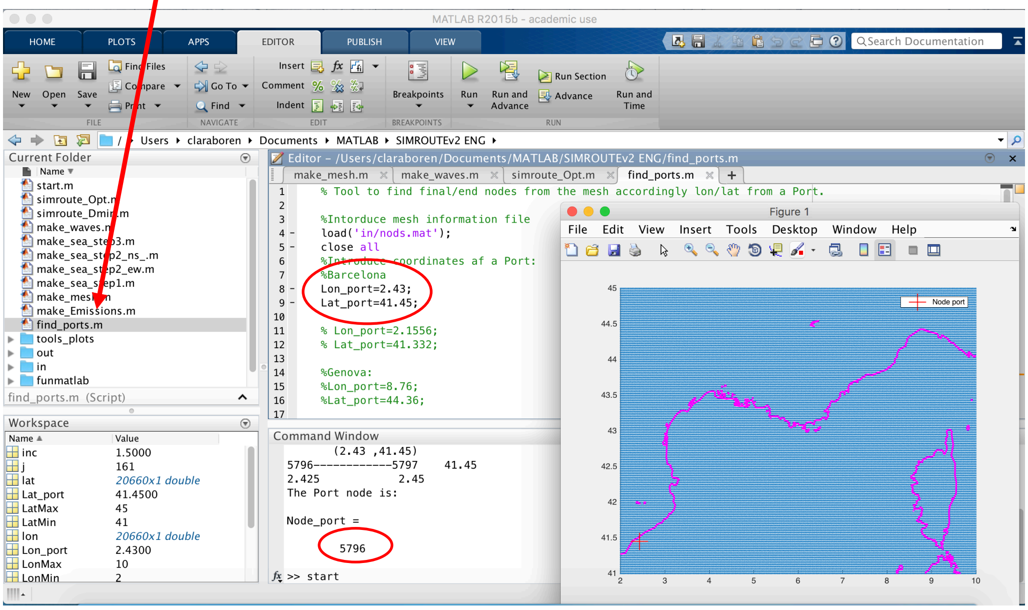

Figure 7: find_ports script to find start node

In the Editor window, the user has to introduce the longitude and latitude for the port of departure at first place and RUN the script (see in line 8 and 9 of figure 7). This process will give the node number corresponding to those coordinates in the Command window (see Node_port = 5796, figure 7).

A map of the coast-line showing the location of the port of departure node (nod_ini), shown by a red cross, will be displayed (see figure 7).

The number shown in the Command Window as Node_port is the one the user has to introduce as nod_ini variable in *simroute_Opt.m* script.

The user has to repeat the process with the port of arrival, run the *find_ports* script again and the number given in the Command window is the one to introduce as nod_end in *simroute_Opt.m* script.

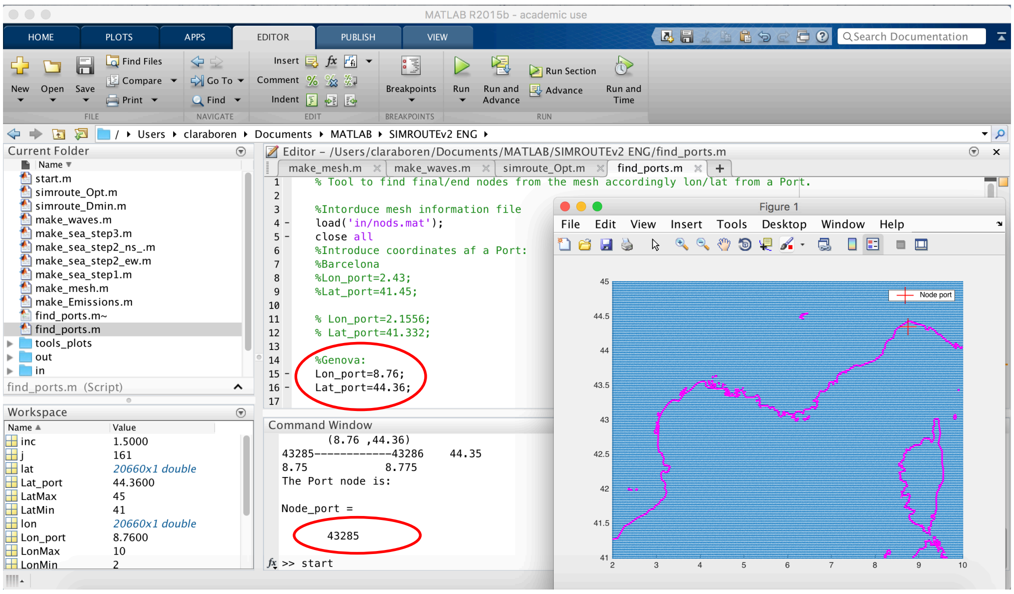

Figure 8: find_ports script to find end node

The node number found, 43285, is the one that has to be introduced as end_node in *simroute_Opt.m* script.

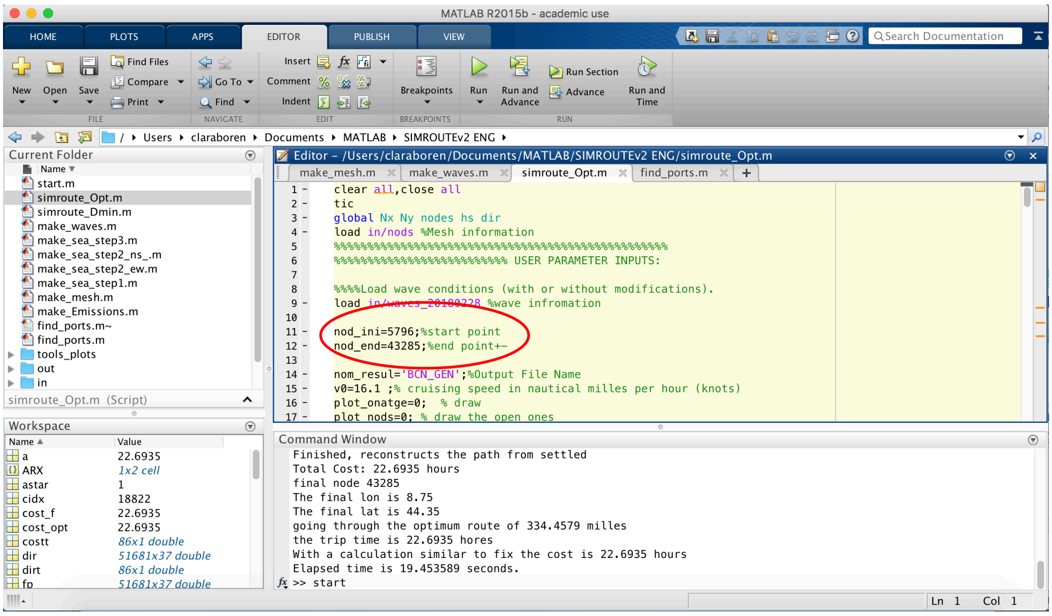

Figure 9: simroute_Opt.m script with start and end nodes introduced

At this point, the user has to RUN *simroute_Opt.m* script and make sure that neither the start node nor the end node are land.

When running *simroute_Opt.m*, if the message "nod_ini is land" or "nod_end is land" appear, the user has to change the node number to the following one and RUN the *simroute_Opt.m* again. The user has to do this process as many times as needed in order to select the closest nodes to the ports coordinates which are located at sea.

Once *simroute_Opt.m* is run, the user can check the value of the variables by double clicking on them in the Workspace window and a new window will be generated, the Variables window (see figure 10).

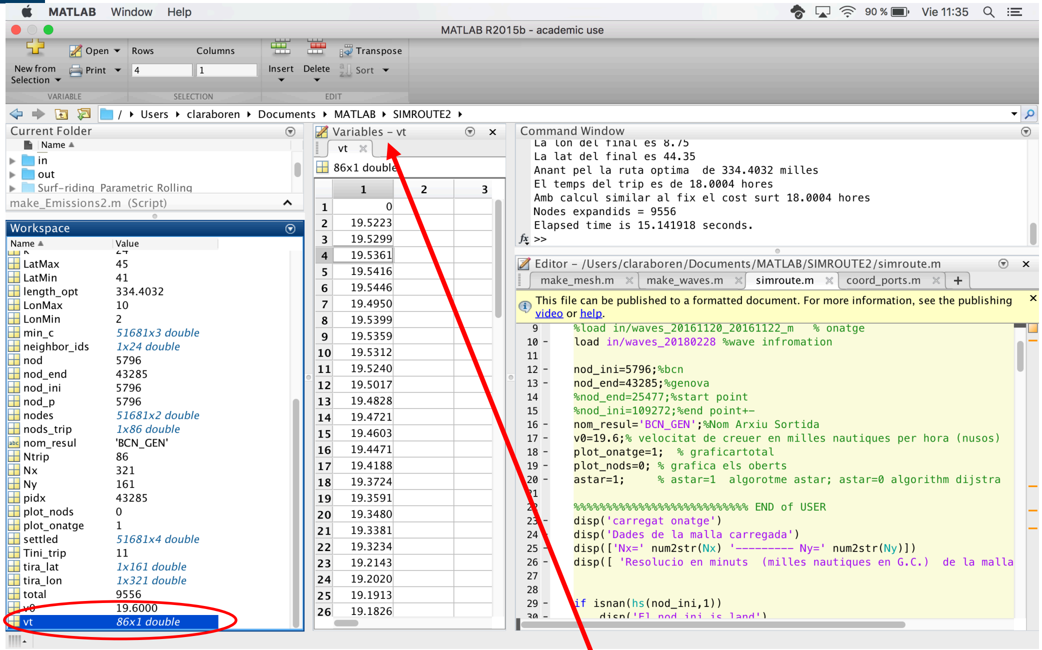

Figure 10: variables shown in Variables window

6) Call *simroute_Dmin.m* in Editor mode by double clicking on it in the Current Folder window.

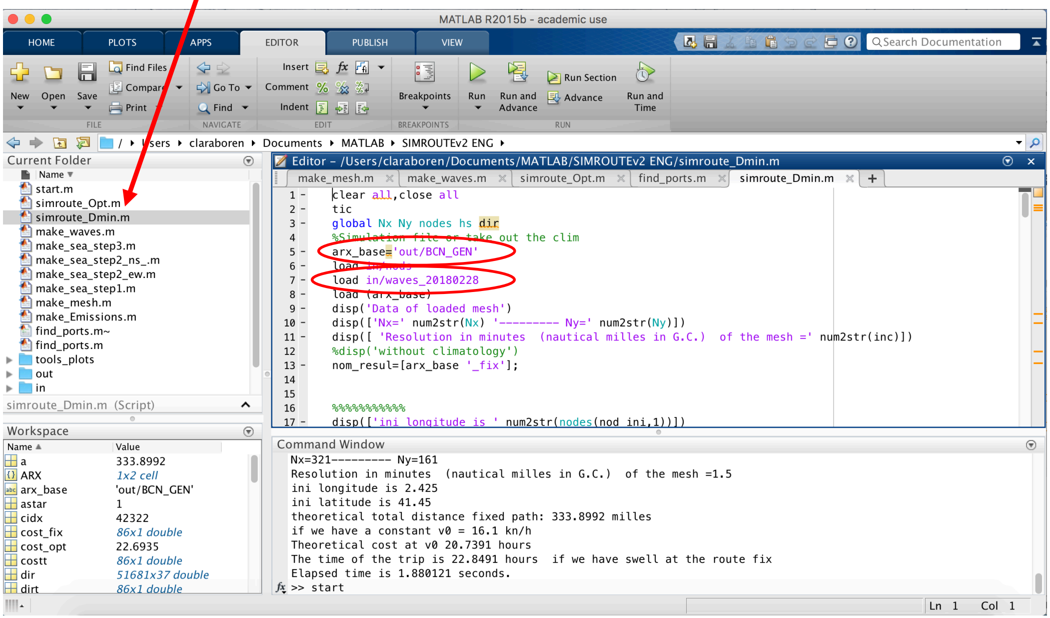

Figure 11: simroute_Dmin script

Simroute_Dmin.m is called in order to calculate the shortest path. In this script, the user has to give a name to the output file and locate it in the OUT folder as shown at line 5 of figure 11.

arx_base='out/BCN-GEN'

The user has also to ask the script for loading the interpolated waves file (see line 7 of figure 11):

load in/waves_20180228

At this moment, the user can RUN the *simroute_Dmin.m* script.

7) Call *plot_routes.m* on Editor mode by double clicking on it in Current Folder window.

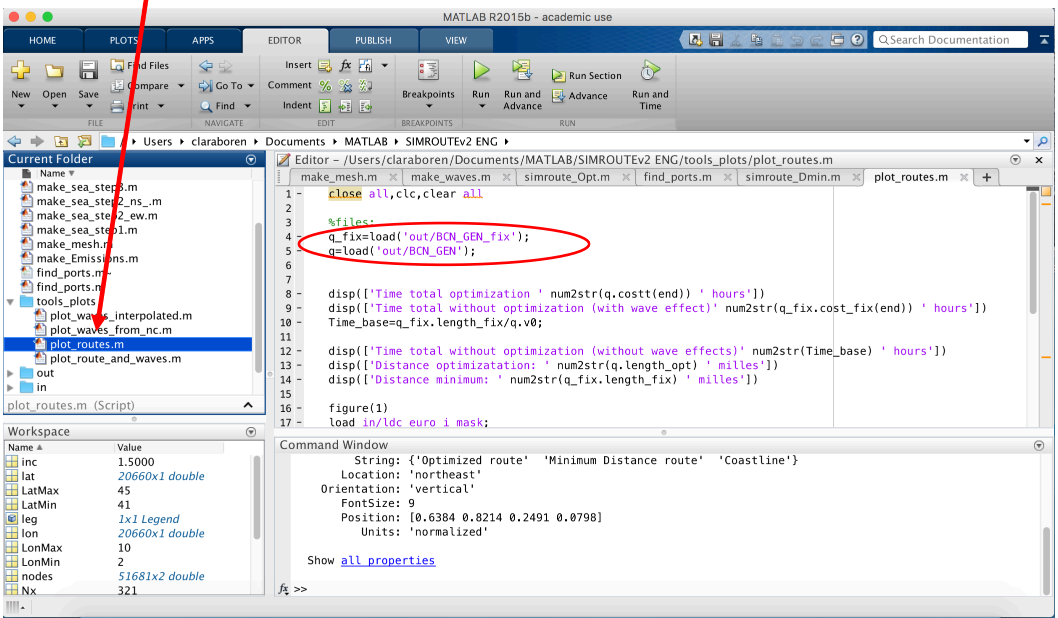

Figure 12: plot_routes.m script

On this script, the user has to ask for loading the output files of *simroute_Opt.m* and *simroute_Dmin.m* (see lines 4 and 5 of figure 12, respectively):

```
q_fix=load('out/BCN_GEN_fix');
q=load('out/BCN_GEN');
```

At this point the user has to RUN *plot_routes.m* and the optimum and minimum distance routes will be plotted.

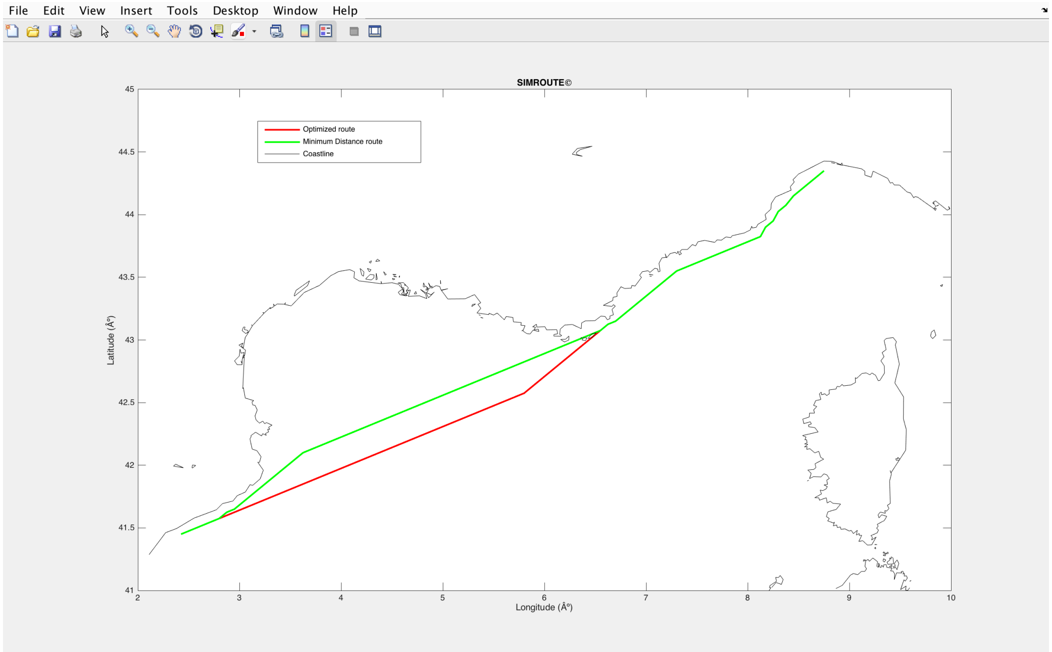

Figure 13: optimum and minimum distance routes plotted

8) *plot_routes_from_nc.m*

   The user could call at this point the function *plot_routes_from_nc.m* and RUN it in order to see the wave field taken from the *netcdf* file downloaded. (see figure 14 and 15).

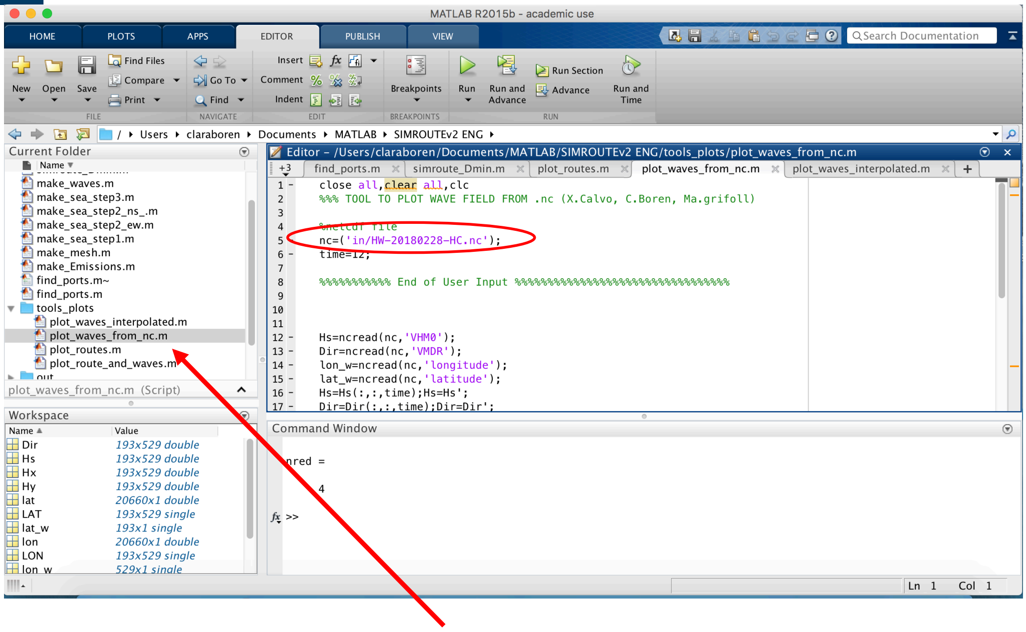

Figure 14: plot_routes_from_nc.m

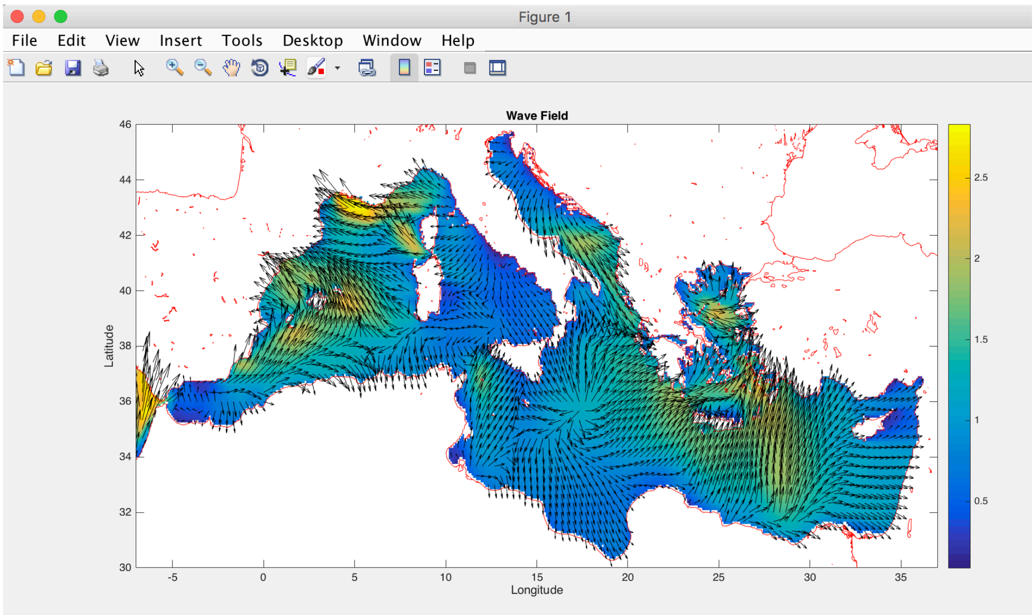

Figure 15: plot_routes_from_nc.m (waves height and direction)

9) *plot_routes_interpolated*

The function *plot_routes_interpolated.m* can be called and RUN in order to see the interpolated wave fields (see figure 16 and 17).

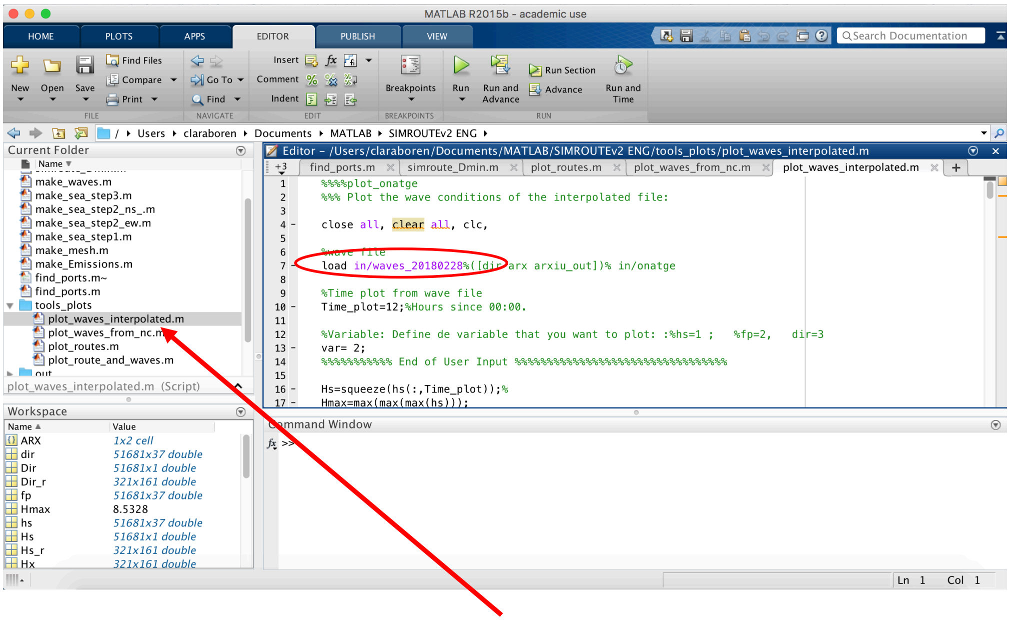

Figure 16: plot_routes_interpolated.m

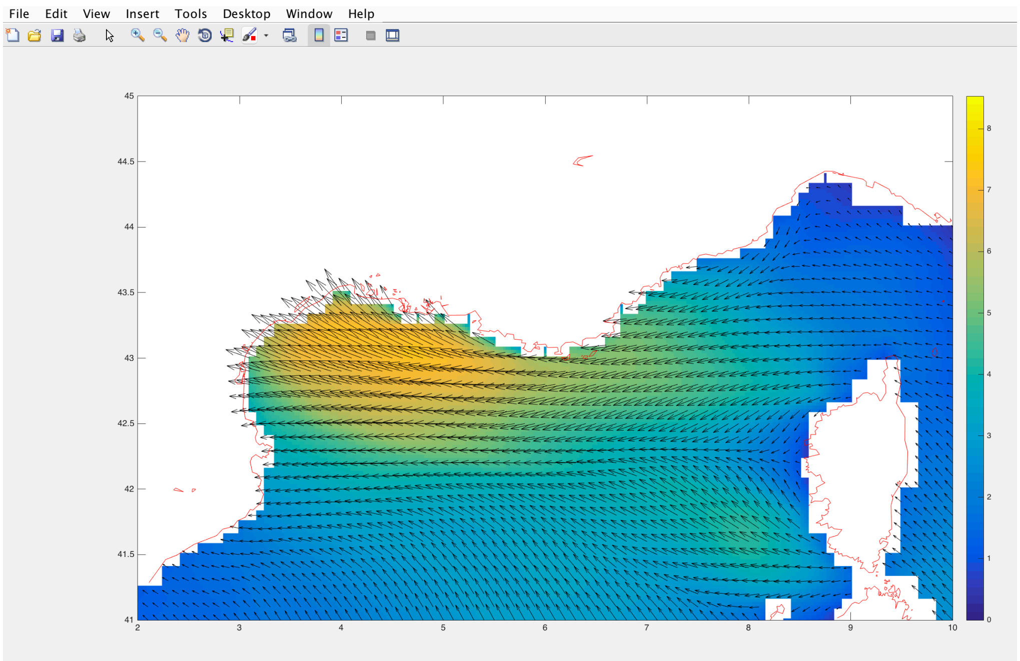

Figure 17: plot_routes_interpolated.m

10) plot_route_and_waves

In order to see the evolution of the storm and the optimized and the minimum distance route in an hourly manner, the user has to call the function *plot_routes_and_waves*. Below are shown some plots of the Test Case as presented above (see figure 18).

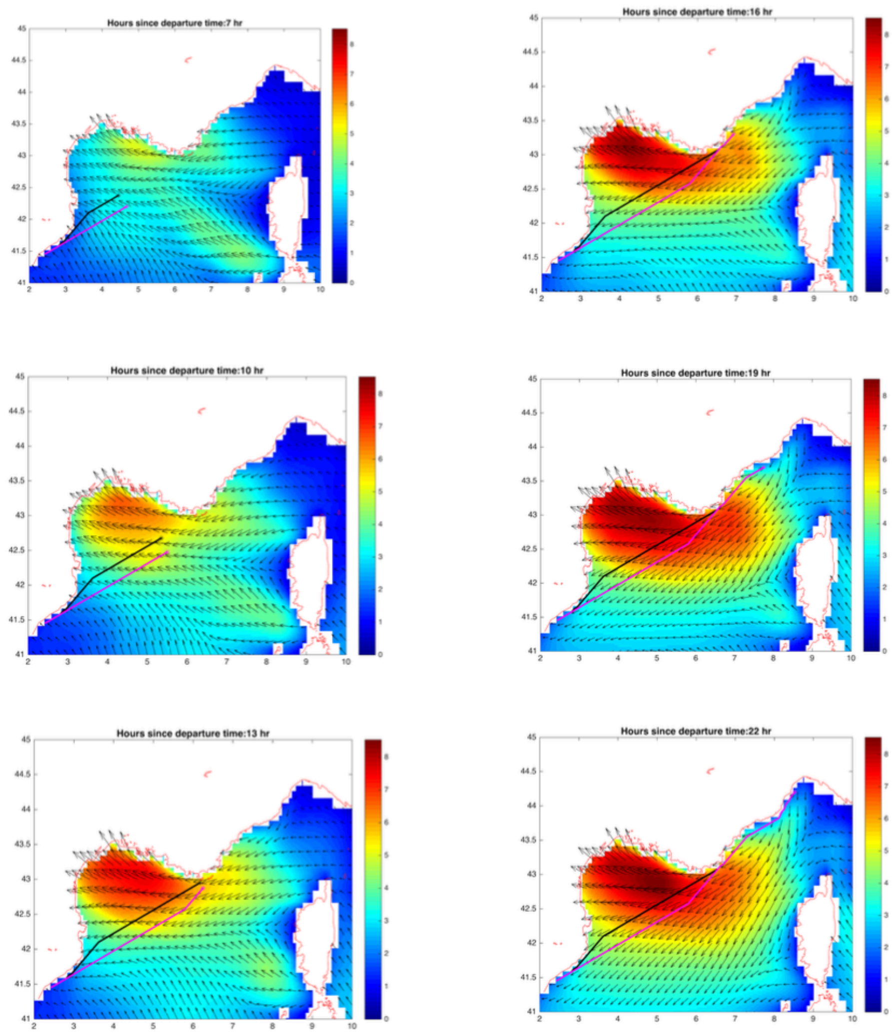

Figure 18: plot_routes_and_waves.m (minimum distance route in black, optimized route in magenta)

The different plots will be saved in folder PLOTS located into the folder OUT.

Before or after plotting, the user can call the module for calculating pollutants emission during the analyzed trip.

11) Call *make_Emissions.m* in Editor mode by double clicking on it on the Current Folder window.

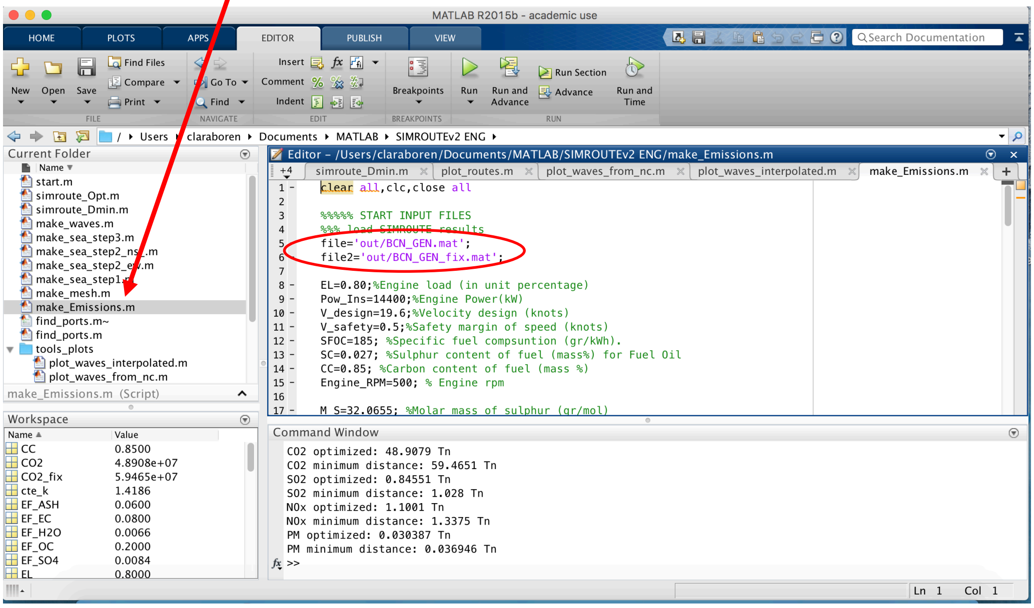

Figure 19: make_Emissions.m

This script calculates the amount of carbon dioxide (CO_2), sulphur dioxide (SO_2), particulate matter (PM) and nitrogen oxides (NOx) emitted into the atmosphere per engine during the trip analyzed.

The required input variables, which can be changed by the user, are the following:

EL=0.80;	Engine load (in unit percentage)
Pow_Ins=14400:	Engine Power (kW)
V_design=19.6;	Velocity design (knots)
V_safety=0.5;	Safety margin of speed (knots)
SFOC=185;	Specific fuel consumption (gr/kWh).
SC=0.027;	Sulphur content of fuel (mass%) for Fuel Oil
CC=0.85;	Carbon content of fuel (mass %)
Engine_RPM=500;	Engine rpm

Additionally, there are some constants also required:

M_S=32.0655;	Molar mass of sulphur (gr/mol)
M_SO2=64.06436;	Molar mass of sulphur dioxide g/mol (number of mols of S=number of mols of SO2)
M_C=12.01;	Molar mass of carbon (gr/mol)
M_CO2=44.0886;	Molar mass of carbon dioxide (gr/mol) (number of mols of C=number of mols of CO2)

EF_EC=0.08	Emission factor elementary carbon (gr/kWh)
EF_OC=0.2	Emission factor for organic carbon (gr/kWh)
EF_ASH=0.06	Emission factor for ash (gr/kWh)
OC_EL=1.024	Part of organic carbon depending on EL (dimensionless)

The user has to ask the program to load the *simroute_Opt.m* and *simroute_Dmin.m* results (see line 5 and 6 figure 19):

```
file='out/BCN_GEN.mat';
file2='out/BCN_GEN_fix.mat';
```

Once the user has checked that all variables are well introduced, he/she has to RUN the *make_Emissions.m* script and the quantity of each pollutant emitted into the atmosphere for the optimized route and for the minimum distance route will appear on the Command Window (see figure 20).

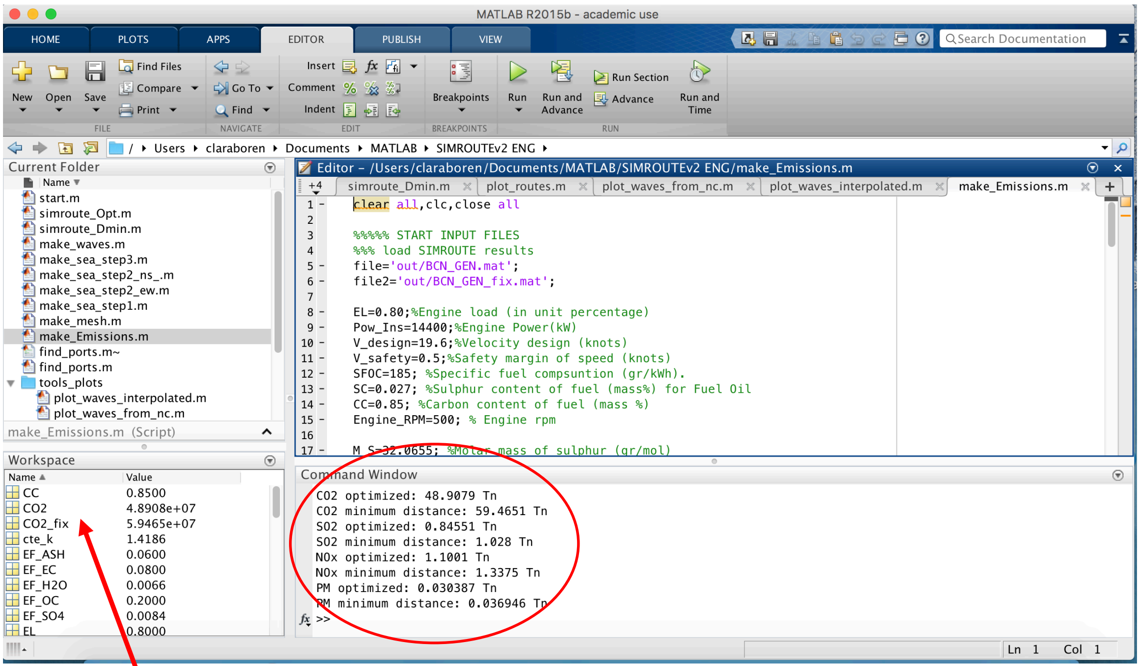

Figure 20: make_Emissions.m results

As already stated above, the user can also click on a variable in order to display the Variable window and see the hourly evolution of that variable, but only when the variable's value changes as a time function (sub index _fix is for the minimum distance route).

As shown in the image above, the quantitative difference in pollutants emitted into the atmosphere between the optimized and the minimum distance route is about 20% more than in the case where the route was not optimized.

The results of above Test Case have been obtained using A* algorithm due to the significant reduction of computational time compared to the Dijkstra model. This reduction in time occurs because less nodes are considered in comparison to the Dijkstra algorithm.

SIMROUTE TECHNICAL MANUAL

SIMROUTE: A WEATHER SHIP ROUTING SOFTWARE FOR ACADEMIC PURPOSES

SIMROUTE®

Software

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Supported by THE NIPPON FOUNDATION

Table of contents

SIMROUTE Technical Manual **4**

1. Basis and Algorithm ***4***

2. Waves ***6***

a) Following seas **7**

c) Head seas **8**

3. Commands ***8***

4. Emissions of pollutants ***9***

a) The initial input data **9**

b) Engine power estimation **10**

c) Emission Factors calculation **12**

Table of figures

FIGURE 1: SAMPLE OF A NODE GRIDDED MESH 4
FIGURE 2: SCHEME OF THE GRID RESOLUTION, 16 EDGES PER NODE 6
FIGURE 3: TYPICAL SPEED REDUCTION CURVES (PADHY, 2008) 8

SIMROUTE Technical Manual

1. Basis and Algorithm

Pathfinding or pathing is the plotting, by a computer application, of the shortest route between two points.

The two well-known pathfinding algorithms normally used are the Dijkstra Algorithm (Dijkstra, 1959) and the A* Algorithm (Dechter and Pearl, 1985). Both of them have been tested and configured in the SIMROUTE.

These pathfinding algorithms are represented through gridded meshes. The meshes are made by different points (nodes). Every single node is separated by the same distance, both horizontal and vertical axes, from the neighbouring node. Therefore, a gridded mesh is built like the following one (Grifoll et al., 2016):

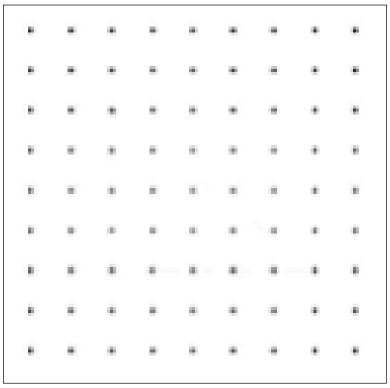

Figure 1: Sample of a node gridded mesh

To each connection (edge) a weight related to the distance is established. The great circle (orthodromic) distances are used for the spherical coordinates of the grid nodes, the same as for transoceanic pathfinding algorithms.

On the one hand, the Dijkstra Algorithm in gridded meshes picks the unvisited vertex with the lowest distance, calculates the distance through it to each unvisited neighbour and updates the neighbour's distance if smaller. Dijkstra algorithm has been used previously in ship routing applications (Mannarini et al., 2013; Montes, 2005).

On the other hand, A* algorithm finds solutions by looking at all the possible path combinations to the result (goal) for the one that obtains the minimum cost (shortest time, shortest distance travelled, etc..) and among these paths it selects the ones that appear to be the fastest to the solution. It is depicted in terms of weighted graphs: beginning from a particular node of a graph, it creates a tree of paths, expanding all these paths one node at a time, until one of the paths reaches a predestined goal node. At each repetition of its main loop, A* algorithm needs to establish which of its unfinished paths to expand into one or more longer paths (Grifoll et al., 2016).

$$f(n) = g(n) + h(n) \qquad (1)$$

Where "n" is the last node on the path, g(n) is the cost of the path from the start node to "n", and h(n) is a heuristic that estimates the cost of the cheapest path from n to the goal. The heuristic is problem-specific. For the algorithm to find the actual shortest path, the heuristic function must be admissible, meaning that it never overestimates the actual cost to get the nearest goal node. In the case of the study, the heuristic function is the minimum distance between origin and destination (Grifoll et al., 2016).

In the function of the grid resolution, path connection options between nodes may vary. Consequently, the sequence of edges followed by the shortest path will be limited by the grid resolution and the connected nodes. As seen in *Figure 2*, edges are connecting nodes displayed by arrows. Every single arrow represents different potential ship courses or directions. Grid resolution can vary according to the investigator's preferences. In this work, different grid resolutions have been tested which obtained similar conclusions to that of Mannarini et al. (2013), which posited that at least 16 edges are required in order to be precise (Grifoll et al., 2016).

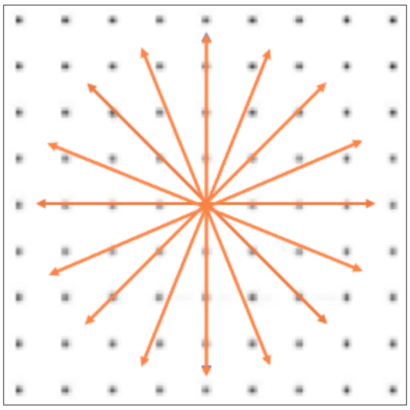

Figure 2: Scheme of the grid resolution, 16 edges per node

2. Waves

Ships can experience challenging operating conditions due to a constant action of waves. Complete knowledge of the expected sea state is important when considering vessel safety.

Wave action is the major factor that affects the ship performance and safety navigation (Hu et al., 2014). Wave fields affect ship motions by decreasing the propeller thrust and adding a resistance in comparison to absence of waves. A simple formula to include ship speed reduction to waves is suggested by Bowditch (2002). The final speed is computed as a function of the non-wave affected speed (v_0) plus a reduction in function of the wave parameters (Grifoll et al., 2016):

$v(H,\Theta) = v_0 - f(\Theta) \cdot H^2$ (2)

Where H is the significant wave height and f is a parameter in function of the relative ship wave direction. The values of f coefficient are shown in Table 1 (Grifoll et al., 2016).

Looking at *Equation (2)*, one can see that the f coefficient has a significant impact on the speed of the vessel. The speed varies depending upon the wave height as well as the wave direction.

The Bowditch formula fits in the algorithm and it subtracts the speed of the vessel with the resistance of the wave in terms of height and direction.

Ship-wave relative direction	**Wave direction**	**f (in kn/ft^2)**
0º≤Θ≤45º	Following seas	0.0083
45º<Θ<135º	Beam seas	0.0165
135º≤Θ≤225º	Head seas	0.0248
225º< Θ<270	Beam seas	0.0165
270º≤Θ≤360º	Following seas	0.0083

Table 1: *Values of the f coefficient*

The relation of the wave direction related on ship navigation can be described as follows (Niclasen, 2010):

a) Following seas

There are many factors that can have a negative impact on stability and ship handling when sailing in the same direction as the waves if the waves are high compared to a vessel. The most notorious is broaching, whereby the vessel is turned violently to one side, leaving it broadside to the oncoming waves. The risk of broaching can be reduced by reducing ship speed to a fraction of the wave speed; but this again increases the risk that overtaking waves wash along upper decks from astern without this being noticed by the operators on the bridge.

b) Beam seas

Sailing in beam seas can results in large roll angles and, in extreme conditions, the vessel can capsize.

c) Head seas

Sailing against the waves is in most cases the best way to negotiate a series of large waves, but this also inflicts the most violent forces on the vessel, increasing the danger of slamming and shifting of cargo. The impact forces can be limited, to some extent, by reducing the vessel speed, or altering course.

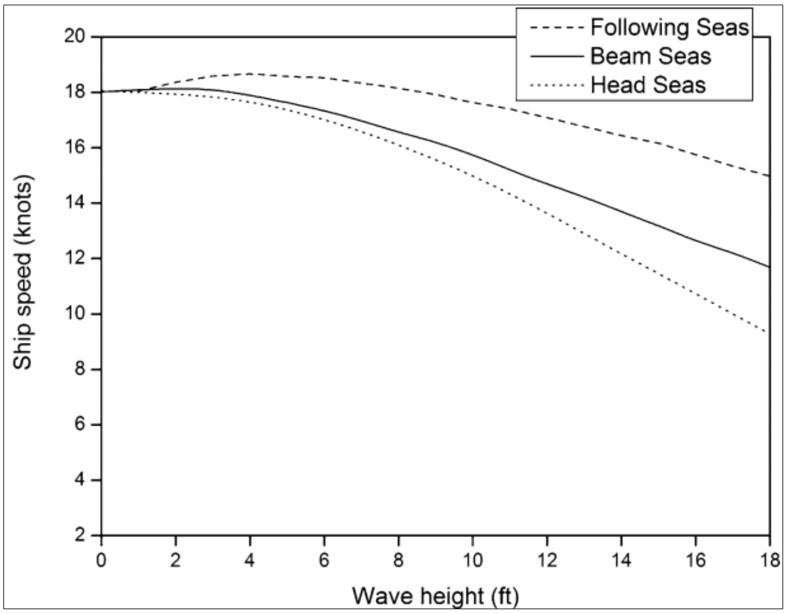

Figure 3: Typical speed reduction curves (Padhy, 2008)

3. Commands

The algorithm in Matlab possesses folders that have been previously created. In the folders, there are different files that have been made with the ".m" format in order to introduce the needed data and make them run in a concrete order. Following this paragraph, the procedure is explained:

Through the "make_mesh.m" file the corresponding latitudes and longitudes are introduced. After running the script within this file, the Western Mediterranean mesh is created.

When looking at the "make_waves.m" file, the use of the correct wave scripts has to be ensured. The wave scripts, taken from the Puertos del Estado website, have to be introduced in the file. Each script poses the waves conditions (wave height and direction) of one entire day. For instance, if the path of the route takes 80 hours and only 3 wave scripts (accounting for 72 hours) have been introduced, the algorithm will have an error. Once the correct wave scripts are introduced and the estimated time of departure (ETD) is established, an output wave script is obtained and is used for running the "sim-route.m" file.

When, "simroute.m" is opened, the output script previously obtained is inserted into the file and then the initial speed of the vessel (v_0) is modified. It has to be taken into account that through the "coord_ports.m" file the initial and end nodes (ports) for each route are created. These nodes are also introduced in the "simroute.m" file. The "simroute.m" is run and the results of the optimum route, in hours, are obtained.

Through the output scripts that "simroute.m" creates, "simroute_fix.m" can be run and the results of the shortest path route with and without waves, in hours, will be obtained (see case test of SIM-ROUTE User's Manual).

4. Emissions of pollutants

This module has been developed for obtaining the amount of sulphur dioxide (SO_2), carbon dioxide (CO_2), nitrogen oxides (NO_x) and particulate matter (PM) generated per trip. The calculation process has been inspired by STEAM 2 methodology (Jalkanen et al., 2009).

The calculation process is described below for better comprehension:

a) The initial input data

- Installed power per engine in kW ("**Pow_Inst**", 14400 kW as default value)
- Engine Load ("**EL**" assumed to be 80% when sailing at cruising speed)
- Design speed in knots ("**V_design**", 25 knots as default value)
- Specific Fuel Oil Consumption in g/kWh ("SFOC", 200g/kWh as default)
- Sulphur Content of fuel in mass percentage ("SC", 0.027 for Fuel Oil outside ECA zones)

- Carbon Content of fuel in mass percentage ("CC", 0.85 for Fuel Oil)
- Engine Revolutions per Minute (Engine_RPM, 500 rpm as default value for medium speed diesel engines)
- Molar mass of Sulphur, Sulphur dioxide, Carbon and Carbon dioxide in g/mol

The specific fuel oil consumption of 200 g/kWh is used for all engines as a default value, even though the user can change this parameter in "make_Emissions" file introducing the new value as "SFOC" variable. Emissions of CO_2 and SO_2 are calculated from the fuel consumption and sulphur content, respectively (Jalkanen et al., 2009).

If engine data is unavailable, the ship is assumed to use a 500rpm medium speed diesel engine by default. Nevertheless, the user can change this parameter in "make_Emissions" file by changing the value of "Engine_RPM" variable.

b) Engine power estimation

The instantaneous power can be evaluated as a function of the vessel's speed (ITTC 1999):

$$P_{transient} = (CF + CR + CA + CAA)\left(\frac{1}{2}v^3 S\right)\frac{1}{\varepsilon_0} \quad (3) \qquad \text{(in SI units)}$$

Where:

CF	→	frictional resistance
CR	→	residual resistance
CA	→	appendage resistance
CAA	→	air resistance
ε_0	→	propulsive coefficient
S	→	wet surface of the ship

As above parameters are hull-specific, they cannot be found in available databases. Jalkanen et al. (2009) propose a straightforward solution assuming that they are ship-specific constants and, therefore, the power can be written as follows:

$$P_{transient} = kv^3 \qquad (4)$$

Where k is:

$$k = \frac{\varepsilon_p * P_{installed}}{(V_{design})^3} \qquad (5)$$

Where:

$P_{installed}$	→	total installed power of main engines (kW)
ε_p	→	engine load at Maximum Continuous Rating (MCR) of main engines
V_{design}	→	design speed (m/s)

Be sure to convert m/s into knots when needed.

With this data, the software calculates the average transient power in terms of the vessel's speed by getting the information from the SIM file (for the minimum distance route) and from the SIM_fix file (for the optimized route), which are loaded at the beginning of the script.

Once the power is calculated, the software displays the Fuel Consumption (FC) for both routes, obtained from applying the formula:

$$FC = P_{Transient} * SFOC * Time$$

Where:

FC	→	Fuel Consumption (in g converted into T for displaying)
$P_{Transient}$	→	Instantaneous Power (in kW)

SFOC → Specific Fuel Oil Consumption (in g/kWh)

Time → Trip duration (in hours)

The procedure for calculating the emissions is described as follows:

On the first hand, the Emission Factors for the different pollutants are calculated.

An emission factor is defined as the average emission rate of a given pollutant for a given source, relative to units of activity. In this case, g/kWh.

Once the emission factors are obtained for the different pollutants, their value is multiplied by the transient power (kW), by the engine load (%) and by the duration of the trip (hours), giving as a result the amount of pollutant emitted into the atmosphere (in g, converted into Tonnes for displaying).

The process of calculation of each emission factor is described hereby.

c) Emission Factors calculation

- SO_2

SFOC = Specific Fuel Oil Consumption (g/kWh)
SC = Sulphur content of fuel (mass %)
M(S) = Molar mass of sulphur (g/mol)
m(S) = mass of sulphur (g)
$M(SO_2)$ = Molar mass of sulphur dioxide (g/mol)
n(S) = number of moles of sulphur (mol)
$n(SO_2)$ = number of mols of sulphur dioxide (mol)
$m(SO_2)$ = mass of sulphur dioxide (g)

The number of mols of sulphur are obtained by dividing the mass of sulphur (g) times its molar mass (g/mol).

$$n(S) = \frac{m(S)}{M(S)} \qquad \text{(mol)}$$

Bearing in mind the sulphur combustion stoichiometric formula, for a mol of sulphur burnt, there will be a mol of sulphur dioxide generated:

$S + O_2 \rightarrow SO_2$, that is $n(S) = n(SO_2)$

Furthermore, in order to find out the mass of sulphur burnt per kW in an hourly basis, the Specific Fuel Oil Consumption of the engine has to be multiplied by the Sulphur Content of the fuel burnt.

$m(S) = SFOC * SC$ (g/kWh)

In order to calculate the SO_2 Emission Factor, the molar mass of SO_2 has to be multiplied by the number of mols of SO_2 and bearing in mind that $n(S) = n(SO_2)$, $n(S) = \frac{m(S)}{M(S)}$ and $m(S) = SFOC * SC$. Therefore,

$$EF(SO_2) = M(SO_2) * n(SO_2) = M(SO_2) * n(S) = M(SO_2) * \frac{SFOC * SC}{M(S)} \quad \text{(g/kWh)}$$

- CO_2

The process is similar to the one followed for calculating the emission of sulphur dioxide.

SFOC = Specific Fuel Oil Consumption (g/kWh)
CC = Carbon content of fuel (mass %)
M(C) = Molar mass of carbon (g/mol)
n(C) = number of moles of carbon (mol)
m(C) = mass of carbon (g)
$M(CO_2)$ = Molar mass of carbon dioxide (g/mol)
$n(CO_2)$ = number of mols of carbon dioxide (mol)
$m(CO_2)$ = mass of carbon dioxide (g)

$$n(C) = \frac{m(C)}{M(C)} = \frac{SFOC*SC}{M(C)}$$

$$n(C) = n(CO_2)$$

$$EF(CO_2) = M(CO_2) * n(CO_2)$$

- NO_x

The program uses crankshaft rpm (rpm, revolutions per minute) data to assign NOx emission factors, which are based on the following (IMO, 1997):

$$\text{NOx Emission Factor (g/kWh)} = \begin{cases} 17\text{, for engines less than 130 rpm} \\ 45.0\text{^}n^{-0,2}\text{, for engines } 130<n<2000\text{, n= engine rpm} \\ 9.8\text{ for engines over 2000 rpm} \end{cases}$$

Therefore, the NOx emission will depend on the speed of rotation of the engine. The default value is 500 rpm. Nevertheless, this value can be changed by the user at any time.

- PM

Particulate matter is divided into Elementary Carbon (EC), Organic Carbon (OC), Ash, Sulphate (SO_4) and associated water (H_2O).

According to Jalkanen (2009), a linear regression to the data presented by Buhaug et al. (2009) can be applied, giving the following emission factor dependencies:

$$EF_{SO_4} = 0.312S$$

$$EF_{H2O} = 0.244S$$

Where S is the fuel sulphur content in percentages.

$$OC_{EF} = \begin{cases} 3.333, & EL < 0.15 \\ \frac{a}{1+be^{-cEL}}, & EL \geq 0.15 \end{cases}$$

Where a, b and c are dimensionless constants (a=1024, b=47600, c=32547).

$$EF_{EC} = 0.08\ g/kWh$$
$$EF_{OC} = 0.2\ g/kWh$$
$$EF_{ash} = 0.06\ g/kWh$$

The emission coefficients for EC, OC and ash have been assumed to be independent of the sulphur content.
For the emission coefficient for OC, an additional dependency on engine load is used.

The total PM emission factor is assumed to be the sum of all above emission factors:

$$EF_{PM} = SFOC_{REL}(EF_{SO4} + EF_{H2O} + EF_{OC}OC_{EL} + EF_{EC} + EF_{ASH})$$

Where:

$$SFOC_{REL} = 0.455EL^2 - 0.71EL + 1.28$$

$$SFOC = SFOC_{REL} * SFOC_{MANUFACTURER}$$

It is assumed that the NOx emission factors of all engines, regardless of their year of construction, can be computed based on the IMO curve and are independent of the fuel consumption. However, the predictions of the emissions of SO_2, CO_2 and PM are based on engine-specific fuel consumption (Jalkanen et al., 2009).

Technical papers

Grifoll, M., Martínez de Osés, F.X., 2016. A Ship Routing System Applied at Short Sea Distances. A: *Journal of Maritime Research*. Vol. XIII, núm. II, p. 3-6.

Grifoll, M., Martínez de Osés, F.X. i Castells, M., 2018. Potential economic benefits of using a weather ship routing system at Short Sea Shipping. A: *WMU Journal of Maritime Affairs* [on-line]. WMU Journal of Maritime Affairs, Vol. 17, núm. 2, p. 195-211. ISSN 1651-436X. DOI 10.1007/s13437-018-0143-6.

Grifoll, M. et al., 2018. Ship weather routing using pathfinding algorithms: The case of Barcelona - Palma de Mallorca. A: *Transportation Research Procedia* [on-line]. Elsevier B.V., Vol. 33, p. 299-306. ISSN 23521465. DOI 10.1016/j.trpro.2018.10.106.

De Osés, X.M., Castells, M., 2009. The external cost of speed at sea: An analysis based on selected short sea shipping routes. A: *WMU Journal of Maritime Affairs*. Vol. 8, núm. 1, p. 27-45. ISSN 16541642. DOI 10.1007/BF03195151.

References

Basiana, L., Castells, M., Grifoll, M., Martínez, F. X., Borén, C., 2017. Ship-weather routing applied to short sea distances: study of the feasibility of SIMROUTEv2 algorithm. *In: International Association of Maritime Universities Annual General Assembly. "IAMU AGA 17 - 18th International Association of Maritime Universities Annual General Assembly"*. Varna, Bulgary. P. 330-340.

Delitala, A.M.S. et al., 2010. Weather routing in long-distance Mediterranean routes. A: *Theoretical and Applied Climatology*. Vol. 102, núm. 1, p. 125-137. ISSN 14344483. DOI 10.1007/s00704-009-0238-2.

Delft, C. et al., 2006. Greenhouse Gas Emissions for Shipping and Implementation Guidance for the Marine Fuel Sulphur Directive. A: [on-line]. p. 276.

Hinnenthal, J. i Clauss, G., 2010. Robust Pareto-optimum routing of ships utilising deterministic and ensemble weather forecasts. A: *Ships and Offshore Structures*. Vol. 5, núm. 2, p. 105-114. ISSN 17445302. DOI 10.1080/17445300903210988.

IMO. (n.d.-a)., 2016. Air Pollution, Energy Efficiency and Greenhouse Gas Emissions.

Jalkanen, J.P. et al., 2009. A modelling system for the exhaust emissions of marine traffic and its application in the Baltic Sea area. A: *Atmospheric Chemistry and Physics*. Vol. 9, núm. 23, p. 9209-9223. ISSN 16807324. DOI 10.5194/acp-9-9209-2009.

Jalkanen, J.P., 2009. A modelling system for the exhaust emissions of marine traffic and its application in the Baltic Sea area. Suplement 1: The procedure for calculating the SO x emission factor from fuel sulphur. A:. Vol. 85, núm. C, p. 1-2. ISSN 0161-8105. DOI 10.1086/599017.

Jalkanen, J.P. et al., 2012. Extension of an assessment model of ship traffic exhaust emissions for particulate matter and carbon monoxide. A: *Atmospheric Chemistry and Physics*. Vol. 12, núm. 5, p. 2641-2659. ISSN 16807316. DOI 10.5194/acp-12-2641-2012.

Johansson, L., Jalkanen, J.P. i Kukkonen, J., 2017. Global assessment of shipping emissions in 2015 on a high spatial and temporal resolution. A: *Atmospheric Environment*. Vol. 167, p. 403-415. ISSN 18732844. DOI 10.1016/j.atmosenv.2017.08.042.

Mannarini, G. et al., 2013. A Prototype of Ship Routing Decision Support System for an Operational Oceanographic Service. A: *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation* [on-line]. Vol. 7, núm. 2, p. 53-59. ISSN 2083-6473. DOI 10.12716/1001.07.01.06.

Padhy, C.P., Sen, D. i Bhaskaran, P.K., 2008. Application of wave model for weather routing of ships in the North Indian Ocean. A: *Natural Hazards*. Vol. 44, num. 3, p. 373-385. ISSN 0921030X. DOI 10.1007/s11069-007-9126-1.

Panigrahi, J.K. et al., 2012. Optimal ship tracking on a navigation route between two ports: A hydrodynamics approach. A: *Journal of Marine Science and Technology*. Vol. 17, num. 1, p. 59-67. ISSN 09484280. DOI 10.1007/s00773-011-0116-3.

Prpić-Oršić, J. et al., 2014. Influence of ship routes on fuel consumption and CO2 emission. A: *Maritime Technology and Engineering* [on-line]. September, p. 857-864. DOI 10.1201/b17494-114.

Sen, D. i Padhy, C.P., 2010. Devlopment of a Ship weather-routing algorithm for specific application in North Indian Ocean Region. A: *The International Conference on Marine Technology*. Vol. 50, December, p. 21-27.

Simonsen, M. H., Larsson, E., Mao, W. and Ringsberg J. W.,2015. State-of-art within ship routing. *Proceedings ASME, 34th International Conference on Ocean, Offshore and Arctic Engineering.* Volume 3: Structures, Safety and Reliability. Canada

Szlapczynska, J. i Smierzchalski, R., 2009. Multicriteria optimisation in weather routing. A: *Marine Navigation and Safety of Sea Transportation* [en línia]. Vol. 3, núm. 4, p. 423.

Takashima, K., Mezaoui, B. i Shoji, R., 2009. On the Fuel Saving Operation for Coastal Merchant Ships using Weather Routing. A: *The International Journal on Marine Navigation and Safety of Sea Transportation* [en línia]. Vol. 3, núm. 4, p. 401-406. ISSN 2083-6473

Tsujimoto, M, Hinnenthal, J., 2008. Optimum Navigation for Minimizing Ship Fuel Consumption—Investigation of Route, Speed and Seakeeping Performance. *Proceedings of the 6th Osaka Colloquium on Seakeeping and Stability of Ships*, Osaka, Japan, pp. 43–50.

Viana, M. et al., 2013. *Impact of international shipping on European air quality*. ISBN 9789292133573.

Walther, L. et al., 2016. Modeling and Optimization Algorithms in Ship Weather Routing. A: *International Journal of e-Navigation and Maritime Economy* [on-line]. Elsevier B.V., Vol. 4, p. 31-45. ISSN 24055352. DOI 10.1016/j.enavi.2016.06.004.

Wei, S. i Zhou, P., 2012. Development of a 3D Dynamic Programming Method for Weather Routing. A: *International Journal on Marine Navigation and Safety of Sea Transportation*. Vol. 6, núm. 1, p. 79-85.

Winther, M. et al., 2016. EMEP/EEA Air Pollutant Emission Inventory Guidebook 2016. A: . núm. 21, p. 1-52. DOI 10.2800/247535.

Zahng, J. and Huang, L., 2016. Optimal Ship Weather Routing using Isochrone Method on the Basis of the Weather Changes. *First International Conference on Transportation Engineering*. Chengdu, China.

Zhu, X. et al., 2016. Ship weather routing based on modified Dijkstra algorithm. A: Mmebc, p. 696-699.

Annex 5. AIS data implementation to a weather ship routing. IAMUS 2018 Xavier Calvo Surià's paper

AUTOMATIC IDENTIFICATION SYSTEM DATA IMPLEMENTATION TO A WEATHER SHIP ROUTING

X. CALVO[*], M. GRIFOLL[†], F. X. MARTÍNEZ DE OSÉS[*], M. CASTELLS[*]

[*] Department of Nautical Science and Engineering
Barcelona School of Nautical Studies (FNB)
Universitat Politècnica de Catalunya – BarcelonaTech
Pla de Plau, 18, 08003 Barcelona, Spain
e-mail: xavi.calvo@hotmail.com
e-mail: fmartinez@cen.upc.edu, http://futur.upc.edu/FranciscoJavierMartinezDeOses
e-mail: mcastells@cen.upc.edu, http://futur.upc.edu/MarcellaCastellsSanabra

[†] Department of Civil and Environmental
Barcelona School of Nautical Studies (FNB)
Universitat Politècnica de Catalunya – BarcelonaTech
Pla de Plau, 18, 08003 Barcelona, Spain
e-mail: manel.grifoll@upc.edu, http://futur.upc.edu/ManelGrifollColls

Keywords: Weather ship routing, AIS, optimum routes, Short Sea Shipping.

Abstract. This contribution uses real data from SSS (Short Sea Shipping) routes to compare them with routes obtained with ship routing software. The research has been developed using SIMROUTE ship routing software along with new functions implemented on it to analyze real routes from AIS (Automatic Identification System) obtained data. In addition, high-resolution meteo-oceanographic forecasts is used for this purpose. Ship routing systems have become more important in recent years as there is a higher interest in reducing costs and fuel emissions in Europe. The results show a real need of ship routing software for SSS actual routes. In this case, a few Western Mediterranean routes have been analyzed and have given up to a 2-hour saving in terms of sailing time reductions for real cases. This investigation gives way to new improvements in the SIMROUTE software for new functions for AIS data implementation and new user interfaces.

1 INTRODUCTION

The Short Sea Shipping industry has become an important part inside a European intermodal transport web because it reduces costs and fuel emissions per goods transported. This is achieved with the design of ship route scheduling [1] (Fagerholt and Lindstad 2007; [2] Fagerholt and Ronen 2013), the improvement of ship energy and efficiency [3] (Longva et al. 2010) and weather routing [4] (Simonsen et al. 2015). The introduction of Weather Ship Routing (WSR) and high-resolution weather meteo-oceanographic predictions may increase the efficiency of the SSS activity.

The objective of this work is to analyse the benefits of using ship routing software in real cases. Previous research showed that ship routing in short distance routes produced relevant time savings during strong wave episodes with significant impact in economic costs [5] (Grifoll et al. 2018) comparing with minimum distance routes. This contribution makes a step forward and compares the optimized routes using WSR along with real routes obtained from the Automatic Identification System (AIS) in the NW Mediterranean Sea. The WSR system, called SIMROUTE, is described in detail in [6] Grifoll et al. (2017) and [7] Grifoll et al. (2018).

2 METHODS

2.1 STRONG WAVE EPISODES

The wave predictions used in this investigation are provided by PdE (Puertos del Estado) and AEMET (Spanish Meteorological Agency). They provide wind and wave field forecasts twice a day for the Western Mediterranean Sea. These waves' forecasts come from the HIRLAM model (High Resolution Limited Area Model) which AEMET runs. The weather forecast horizon is 72 hours and provides variables as significant wave height (Hs), wave direction and wave period.

Generally, strong winds are observed over the Ebro Delta from the NW and from the N in the Gulf of Lion [8] (Bunker 1972; [9] Grifoll et al. 2016). In addition, pressure differences between continental land and the low pressures of the Gulf of Lion originate cyclogenesis activity, which derives in strong waves episodes in the NW of the Mediterranean Sea [10] (Lionello et al. 2006).

2.2 AIS OBTAINED ROUTES

To obtain real routes, the Automatic Identification System data was obtained as waypoints (coordinates and time) from www.marinetraffic.com. The waypoint routes correspond to SSS vessels during strong wave episodes, which were predicted by PdE. Extracted data of each waypoint included: latitude (φ, φ_o), longitude (L, L_0), time (hour, day, month and year), speed and course. With waypoint data orthodromic (great circle) calculations between each of them were made to obtain the total distance (d) navigated and the average speed, which was calculated dividing the distance by the total time of the journey. The great circle distance formula was used:

$$\cos(d) = \sin(\varphi_o)\sin(\varphi) + \cos(\varphi_o)\cos(\varphi)\cos(L - L_0) \qquad (1)$$

With origin and destination positions, average speed and wave predictions different cases were studied using SIMROUTE to obtain the optimum routes. These optimized routes are compared to the real ones obtained by AIS systems. Minimum distance routes are also plotted.

Table 1: List of ports of departure

PM: Palma de Mallorca.	FOS: Fòs.	BCN: Barcelona.
PAL: Palermo.	GEN: Genova.	SAV: Savona.

3 RESULTS

Table 2: Comparison between the time obtained in SIMROUTE and the time obtained from observations using AIS.

Route	Days	V_0 (kn)	Optimum time (h)	Real time (h)	Saving (%)
PM-PAL	27/2-1/3	15.0	32.67	33	1.00
BCN-CVT	01/04	26.0	16.82	17	1.07
CVT-BCN	5/3-6/3	23.0	18.77	19	1.21
BCN-SAV	28/02	20.0	15.72	16	1.75
SAV-BCN	30/3-31/3	21.0	15.64	16	2.26
FOS-GEN	28/2-1/3	8.0	26.16	27	3.13
BCN-SAV	21/03	20.0	17.42	18	3.21
CVT-BCN	02/03	25.0	17.26	18	4.11
BCN-GEN	28/2-1/3	19.6	18.02	20	9.92

The results obtained of comparing between WSR and AIS system are presented in the Table 2. The saving in hours varies from 0.3 h to nearly 1.9 h, which makes 1% the minimum saving, and up to 9.9% the maximum time saved obtained using SIMROUTE for real cases. While SIMROUTE operates using a constant speed, actual ships are able to vary their speed which can make major differences when studying a case with a variant sequence of storm. Figures 1, 2, 3 and 4 show different cases where the optimal route and the real route differs, plotted along with the minimum distance. Significant differences are observed in the course, which the ship uses to avoid the storm and how the ship faces wave direction.

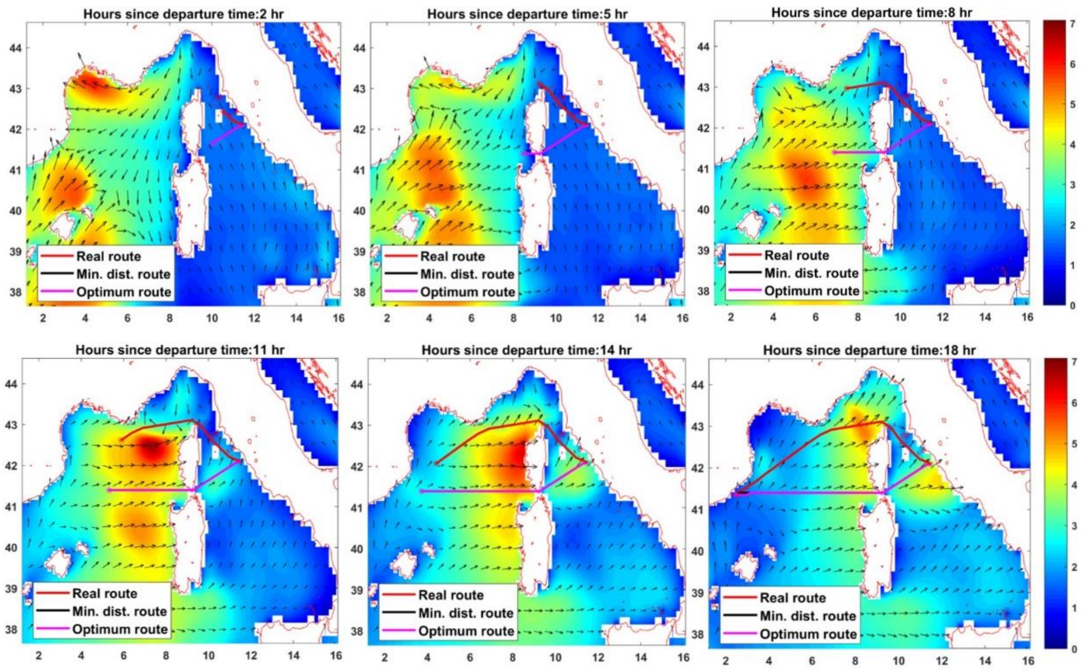

Figure 1: Barcelona - Genova route 28/02-01/3 in 2018, during a storm in the Gulf of Lion, which took 20 hours for Excellent vessel while SIMROUTE took 18 hours. A significant difference in the route can be observed between the optimum (in magenta) and real route (in red), the real ship headed towards Corsica while the WSR followed a more direct route. The minimum distance is also shown in black.

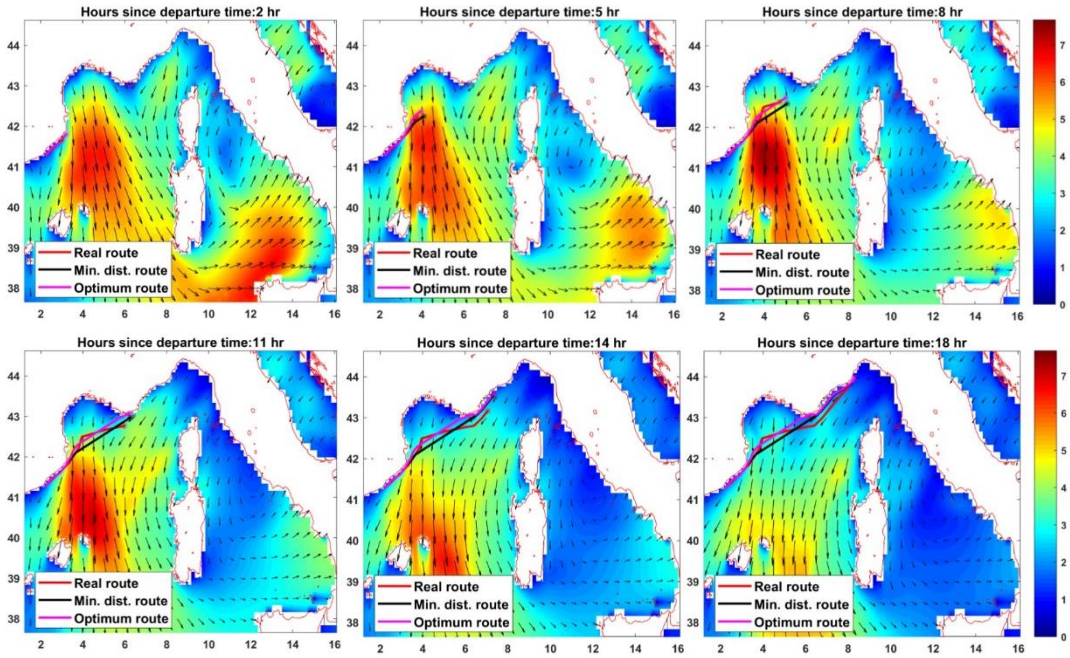

Figure 3: Civitavecchia - Barcelona route on 02/03/2018, with wave episodes concentrated between the Balearic Islands and Corsica and Sardinia, which took 18 hours for Cruise Barcelona vessel while SIMROUTE took 17.2 hours. There is a major difference here between the real route (in red) and the optimum one (in magenta) since the real ship decided to avoid the storm through the N of Corsica while the optimized route stuck to the usual route, very similar to the minimum distance route.

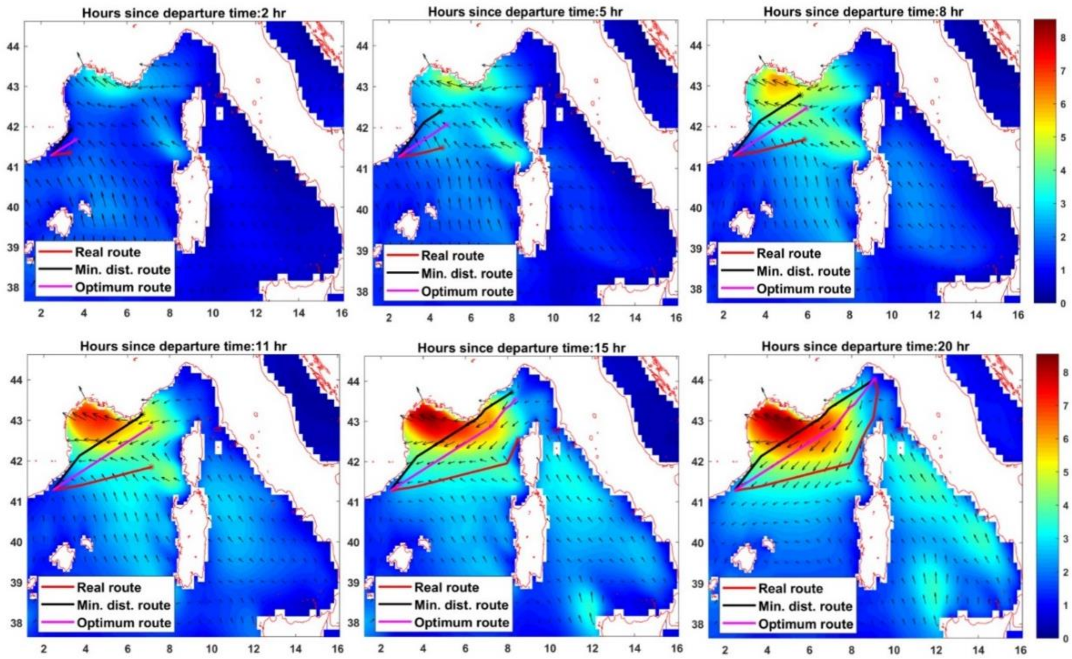

Figure 2: Barcelona - Savona route on 21/03/2018, during a wave episode in front of the NE coast of Spain and NW Mediterranean, which took 18 hours for Cruise Smeralda vessel while SIMROUTE took 17.4 hours. It can be observed that the optimized route (in magenta) is more similar to the minimum distance (in black) while the real route (in red) modifies its course multiple times.

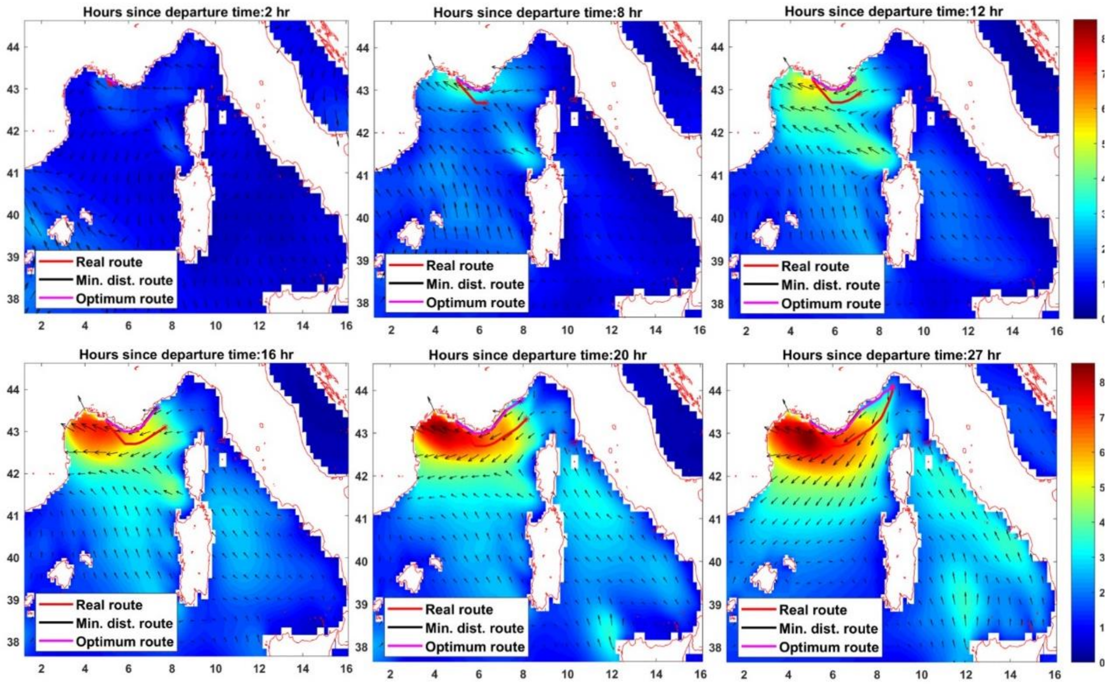

Figure 4: Fos - Genova route 28/2-1/3 in 2018, during a storm in the Gulf of Lion, which took 27 hours for Delta Mariner vessel while SIMROUTE took 26.1 hours. While the real ship (in red) tried to avoid the coast the optimum route (in magenta) navigated closer to the coast. The minimum distance route (in black) is overlapped by the optimum one.

4 CONCLUSIONS

The comparison of AIS data on optimized routes using a WSR system shows a substantial saving of time. A maximum of 1.9 hours of time saved in the BCN-GEN route for a case of 8 m of maximum wave height. The existence of this savings evidences the benefit of WSR systems in Short Sea Shipping activity. Also, it can be observed that ship routing minimizes the time spent on deviating from the main route compared to a real route in an episode of bad weather in Short Sea Shipping routes. The weather ship routing minimizes the time spent on deviating from the main route compared to a real route in an episode of bad weather.

5 REFERENCES

[1] Fagerholt, K., Lindstad, A. TurboRouter: an interactive optimization-based decision support system for ship routing scheduling. In: Maritime Economics & Logistics, 2007, 9(3), 214-233.

[2] Fagerholt, K., Ronen, D. Bulk ship routing and scgheduling: solving practical problems may provide better results. In: Maritime Policy & Management, 2013, 40(1), 48-64.

[3] Longva, T. Eide, MS. Skjong, R. Determining a required efficiency design index level for new ships based on a cost-effectiveness criterion. In: Maritime Policy & Management, 2010, 37(2), 129-143.

[4] Simonsen, MH. Larsson, E. Mao, W. and Ringsberg, JW. State-of-art within ship routing. In: Proceedings ASME, 34th International Conference on Ocean, Offshore and Arctic Engineering. Canada. 2015.

[5] Grifoll, M., Martínez de Osés, F.X., Castells, M. Potential economic benefits of using a weather ship routing system at Short Sea Shipping. In: Journal of Maritime Affairs. World Maritime University, 17(2), 195-211, 2018.

[6] Grifoll, M., Martínez de Osés, F.X. A ship routing system applied at short sea distances. In: Journal of Maritime Research. Vol XIII. No. II 3-6. 2016.

[7] Grifoll, M., Martorell, Ll., Castells, M., Martínez de Osés, F.X. Ship weather routing using pathfinding algorithms: the case of Barcelona – Palma de Mallorca. In: XIII Conference on Transport Engineering. Elsevier. CIT, 2018.

[8] Bunker, AF. Wintertime interactions of the atmosphere with the Mediterranean. In: Journal Physical Oceanography. 1972, 2, 225-238.

[9] Grifoll, M., Castells, M., Martínez de Osés, F.X. Enhancement of Maritime Safety and Economic Benefits of Short Sea Shipping Ship Routing. In: Proceedings of SEAHORSE 2016, International Conference on Maritime Safety and Human Factors. Glasgow. 2016. Available at: http://hdl.handle.net/2117/97774

[10] Lionello, P. Bhend, J. Buzzi, A. Della-Marta, PM. Krichak, S. Jans, ÁA. Maheras, P. Sanna, A. Trigo, IF. Trigo, R. (2006) Cyclones in the Mediterranean region: climatology and effects on the environment: Mediterranean Climate Var: 324-372.